



**Red Hat**

# **CAN'T WE JUST AGREE?**

**ONDŘEJ CHALOUPKA, VOJTĚCH JURÁNEK**

FI MUNI 14. 11. 2019

# WHO WE ARE



WORKING AS ENGINEERS AT RED HAT



DISTRIBUTED SYSTEMS ENTHUSIASTS



MEET US AT BRNO DISTRIBUTED SYSTEMS MEET-UP

<https://www.meetup.com/Brno-Distributed-Systems-Meetup-Group>

# WHAT IS THIS TALK ABOUT



A BRIEF OVERVIEW OF CONSENSUS ALGORITHMS  
IN DISTRIBUTED LEDGER SYSTEMS



WHY CONSENSUS ALGORITHM IS AN ESSENTIAL PART  
IN DISTRIBUTED LEDGER SYSTEMS



HOW THE CHOICE OF CONSENSUS ALGORITHM IMPACTS  
CAPABILITIES OF DISTRIBUTED SYSTEM

# AGENDA



MOTIVATION FOR CONSENSUS IN DISTRIBUTED LEDGER SYSTEM



NON-BYZANTINE FAULT TOLERANT CONSENSUS FOR BLOCKCHAIN



BYZANTINE FAULT TOLERANT CONSENSUS FOR BLOCKCHAIN

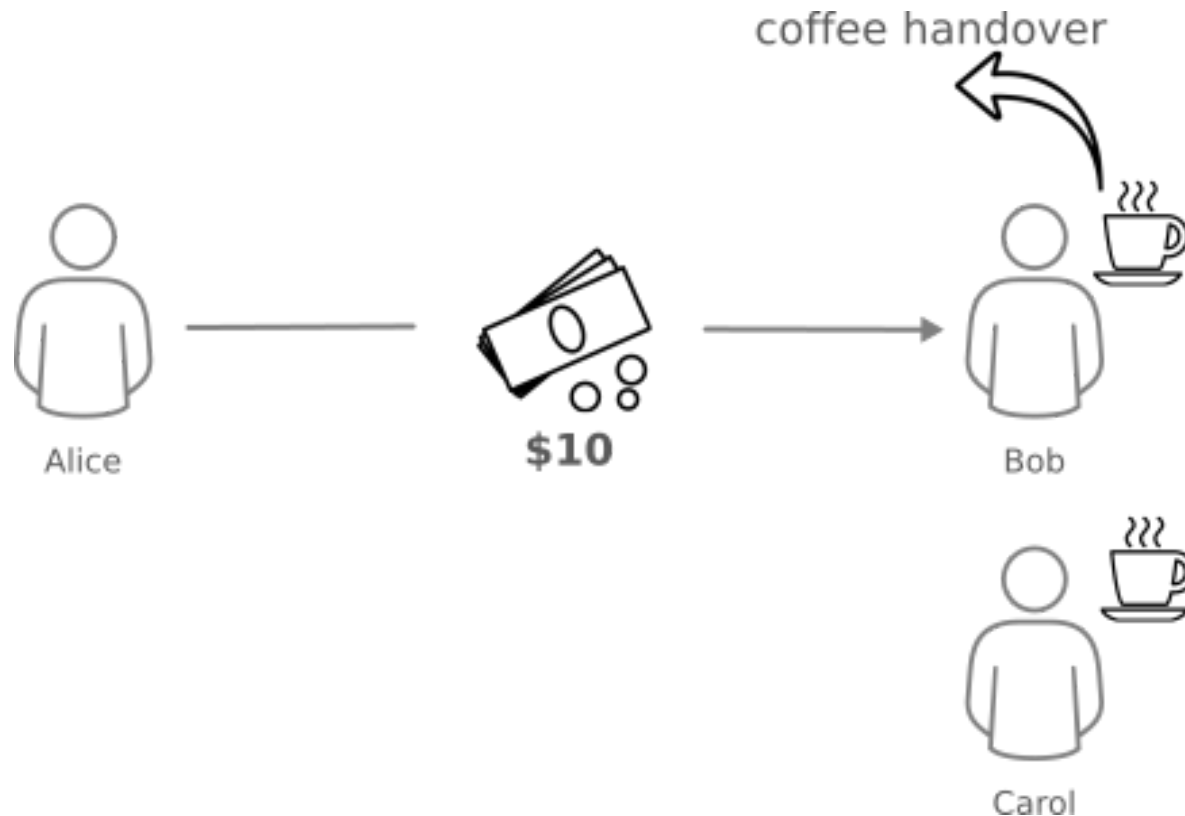


BITCOIN CONSENSUS



CONSENSUS ALGORITHMS ON TOP OF DIRECTED ACYCLIC GRAPHS (DAG)

# TRANSFER OF THE MONEY

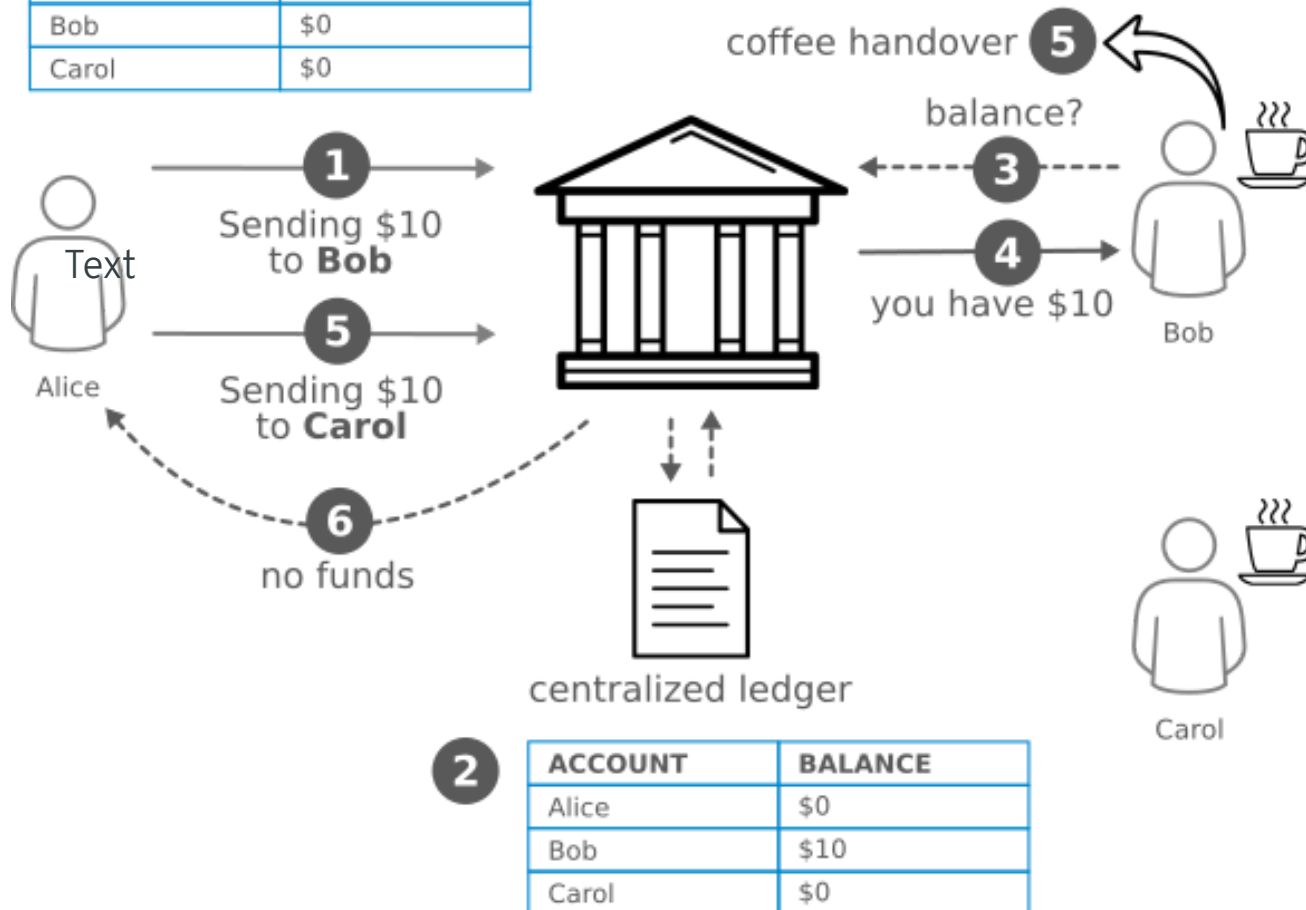


Beware: distributed ledgers are not limited only to cryptocurrencies, there are lots of other applications!

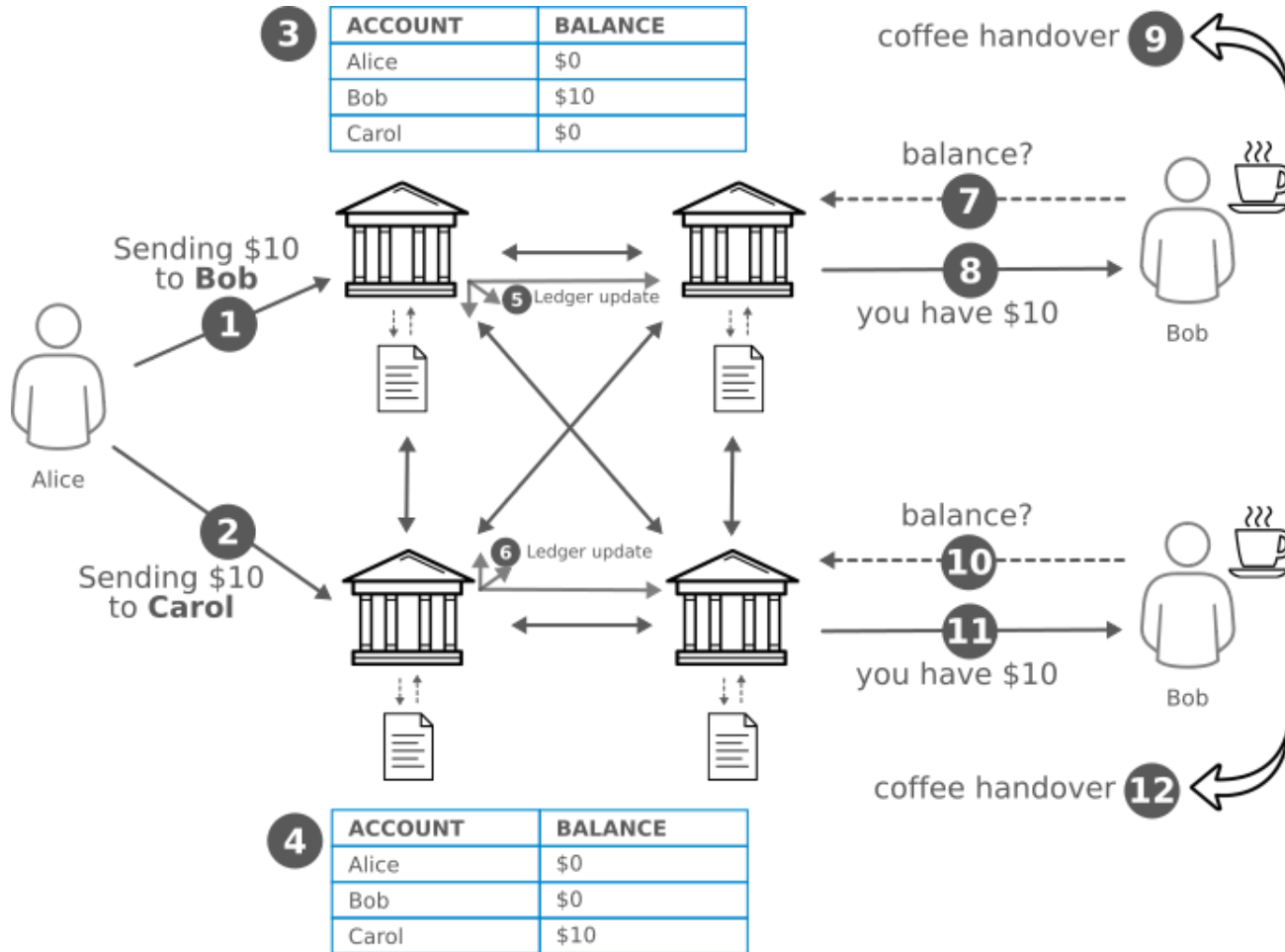
# CENTRAL AUTHORITY

Initial state

ACCOUNT	BALANCE
Alice	\$10
Bob	\$0
Carol	\$0



# DOUBLE SPEND PROBLEM



In distributed systems, we need agreement between participants, which transaction is valid.

# CONSENSUS

- Agreement between nodes on something, e.g. some value
- Another example: whether to commit a (distributed ) transaction to a database
- Hard, network is unreliable
  - delays or failures in communication
- Consensus has to have two properties: **safety** and **liveness**



# CONSENSUS



**Emin Gün Sirer** ✓

@el33th4xor

Follow



Ok, there is a terribly wrong framework emerging around consensus protocols. People think that PoW and PoS are consensus protocols, and that they are the only two consensus protocols out there.

This is false. Let me explain.

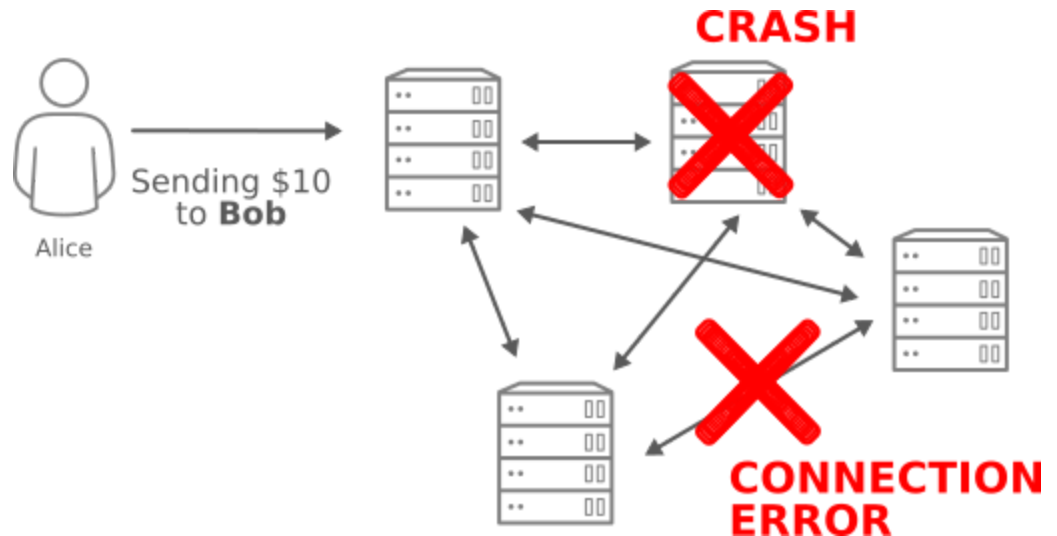
9:09 AM - 13 Jun 2018

<https://twitter.com/el33th4xor/status/1006931658338177024>

# NON-BYZANTINE

## FAULT TOLERANT CONSENSUS

- can withstand failures but not a cheating participant
- trust in all involved parties
- protocols like Paxos (Google Spanner), Raft (etcd), Zab (Zookeeper)
  - leader-based vs. leader-less



# RAFT ALGORITHM

<https://raft.github.io/#raftscope>

# PRIVATE BLOCKCHAINS

- all nodes are under control of a **single organization**
- number of participant is small, they know about each other and **trust each** other
  - Raft in Hyperledger (Sawtooth, on roadmap for Fabric)
- **Permissioned** : nodes needs some permission to join the network
- **Permissionless** : nodes can join the network without any permission (public blockchains)

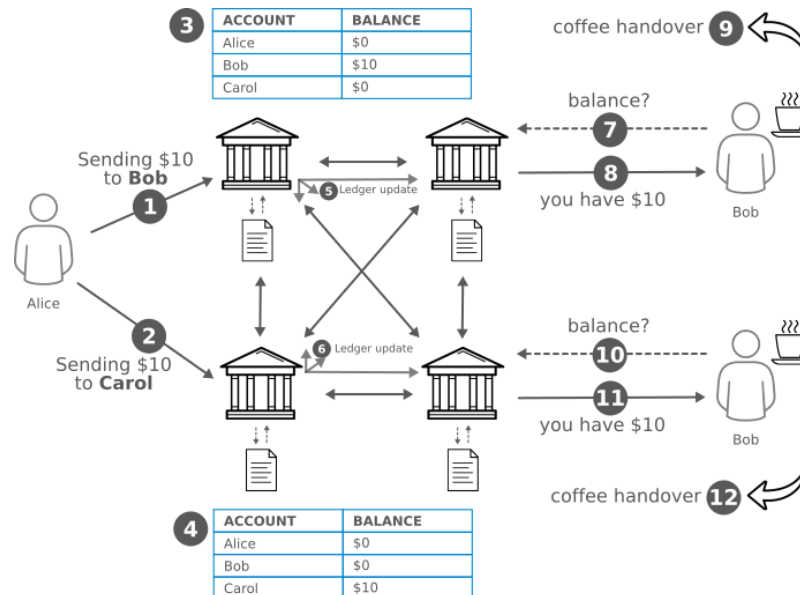
# NON-BYZANTINE CONSENS.

## SUMMARY

- usually used in distributed database systems
- parties do not lie each other
- used in private blockchains
- trust in all involved parties

# BYZANTINE FAILURE

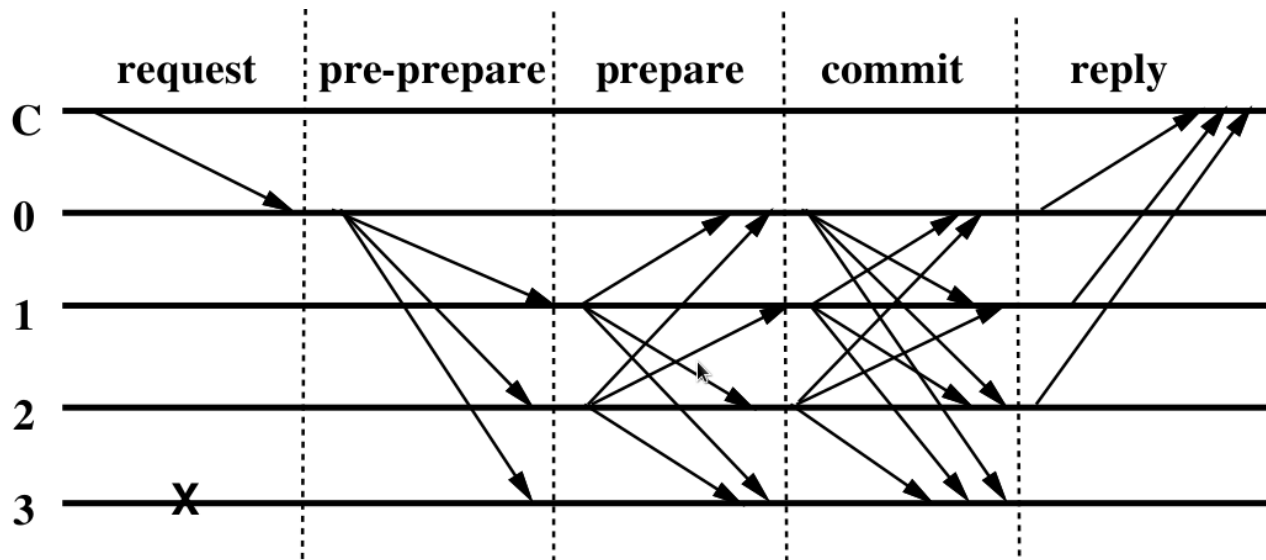
- Besides delays and failures in communication over network, the situation can be even worse - there can be malicious participants!
- A Byzantine failure is any fault presenting different symptoms to different observers.
- E.g. attempt to double spend money.



# BFT ALGORITHMS

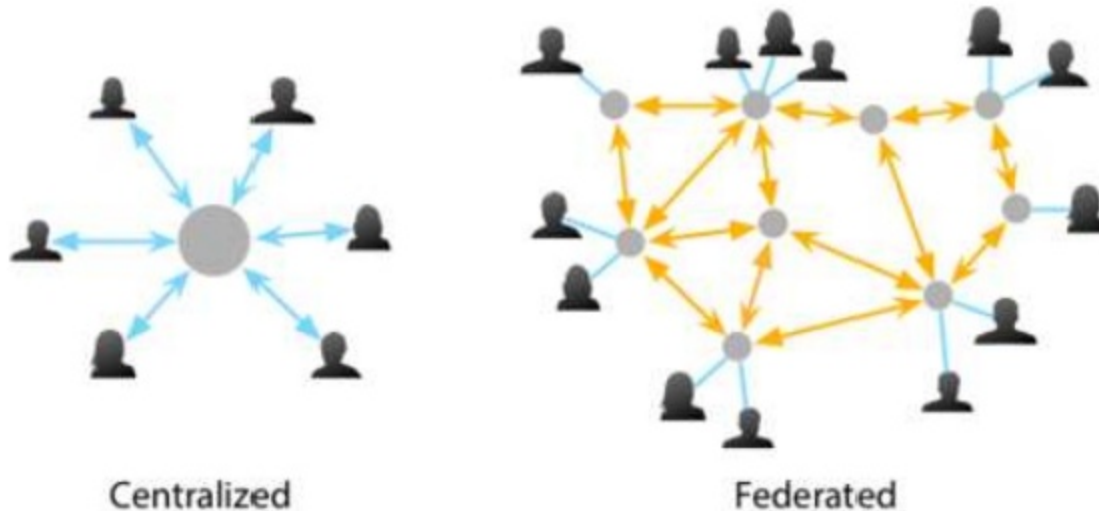
## BYZANTINE FAULT TOLERANT

- PBFT, Tendermint, Stellar ...
- Agreement can be imaged like a three-phase commit (propose a value, pre-prepare commit, prepare, commit)..



# FEDERATED DISTRIBUTED LEDGERS

- Still to some extent centralized.
- Usually to some extent permissioned or have to be combined e.g. with PoS (Tendermint).





# BYZANTINE CONSENSUS

## SUMMARY

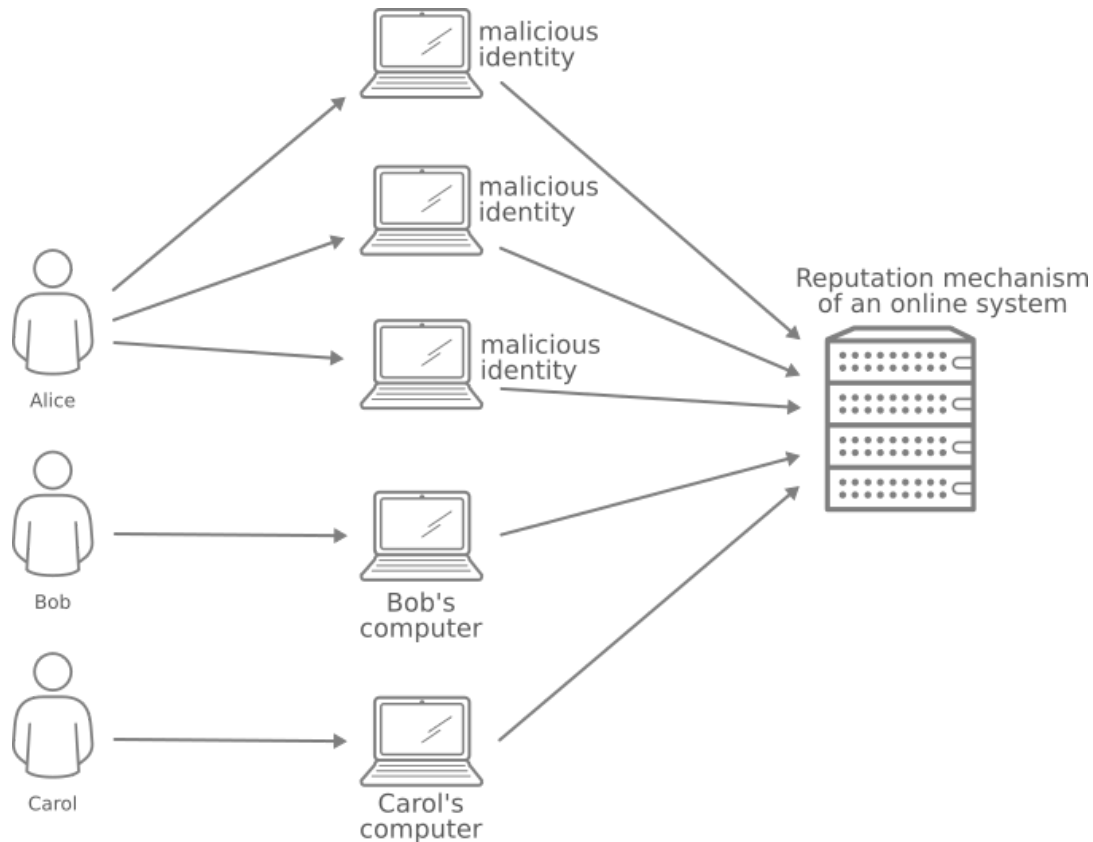
- usually slow
  - a lot of messages exchanged
- parties don't (fully) trust each other
- usually used in semi-public and public blockchains

# PUBLIC BLOCKCHAINS

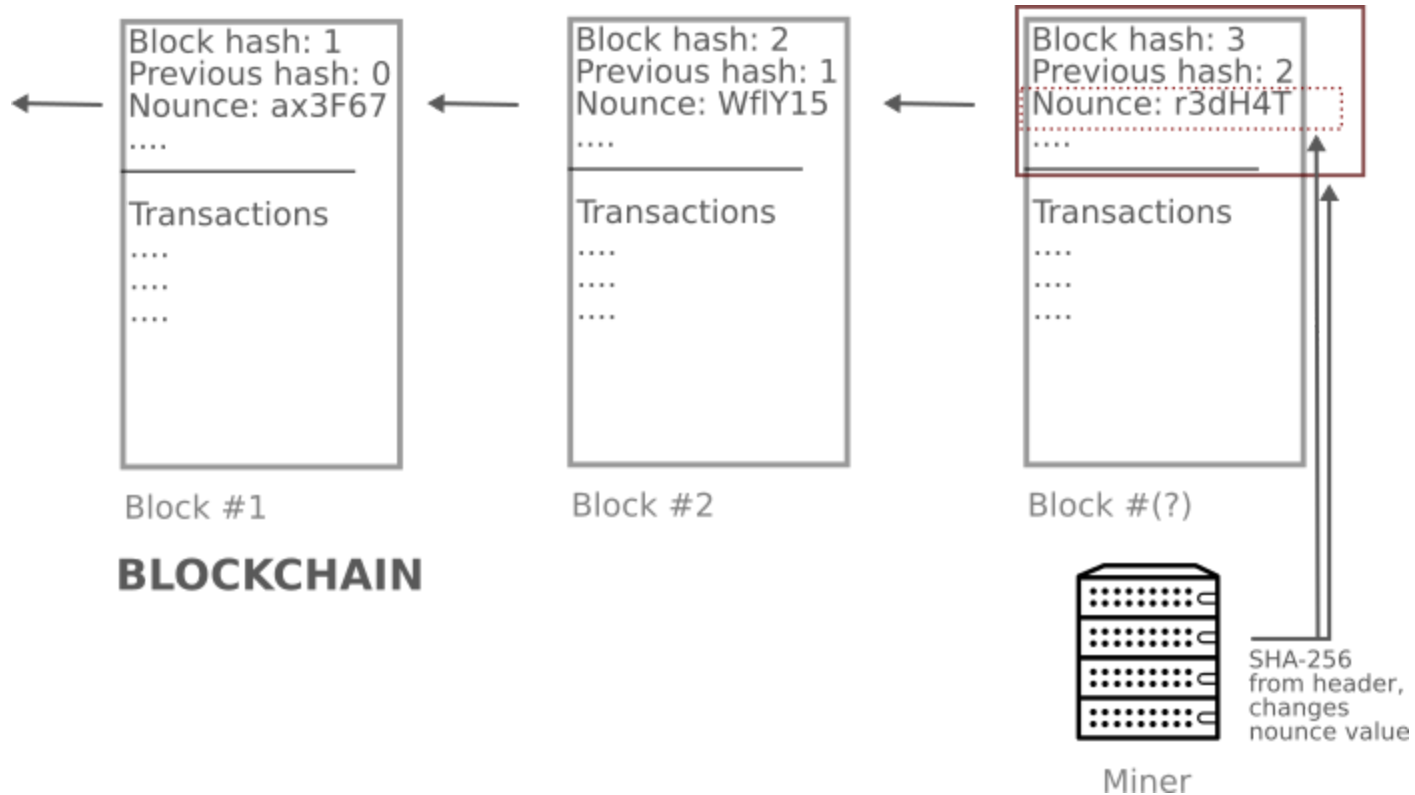
- any party may connect
  - nobody trusts nobody
- some new challenges evolve

# SIBYLS ATTACK

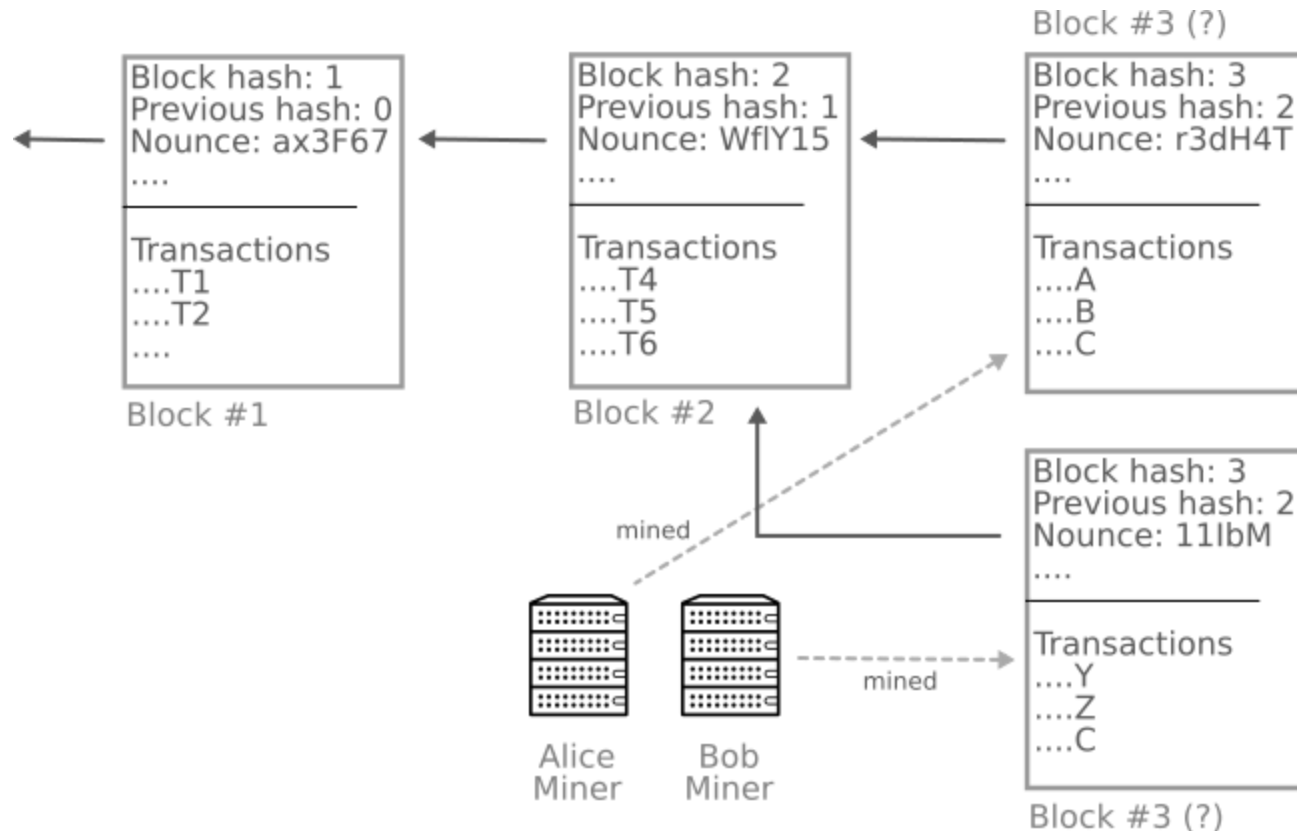
- Forging the identity to subvert the result



# PROOF-OF-WORK



# BLOCKCHAIN FORK



# NAKAMOTO CONSENSUS

- Bitcoin is (almost) **Byzantine fault tolerant (BFT)** and also resistant to **Sybil** attack
- Proof-of-Work (PoW) is mechanism how to prevents Sybil attacks
- The truth (agreement) is determined by the longest chain (created by Proof-of-Work) usually called **Nakamoto consensus algorithm**
- **Probabilistic**: probability of consensus is less than 1
- Proof-of-Work finding a new block is stable to 10 minutes
  - Bitcoin network is essentially **synchronous**

# BITCOIN CONSENSUS

## SUMMARY

- proof-of-work solves sibyls attack
- consensus is a static rule of the longest chain

# WHAT ABOUT POS

## PROOF-OF-STAKE

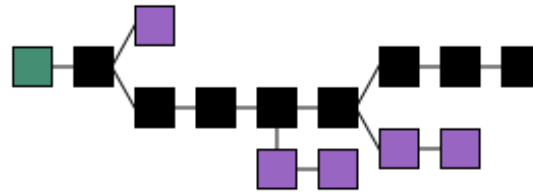
- Proof-of-Stake (PoS) is again **not a consensus** algorithm
- Proof-of-Stake **solves Sybil** attack
- **Chain-based** vs. **PBFT-style**
  - **Hybrid PoS**: Proof-of-Work with Proof-of-Stake on top
- Consensus is solved by a separate rule - e.g. the longest chain wins



# CONSENSUS ALGORITHMS

## ON TOP OF DIRECTED ACYCLIC GRAPHS

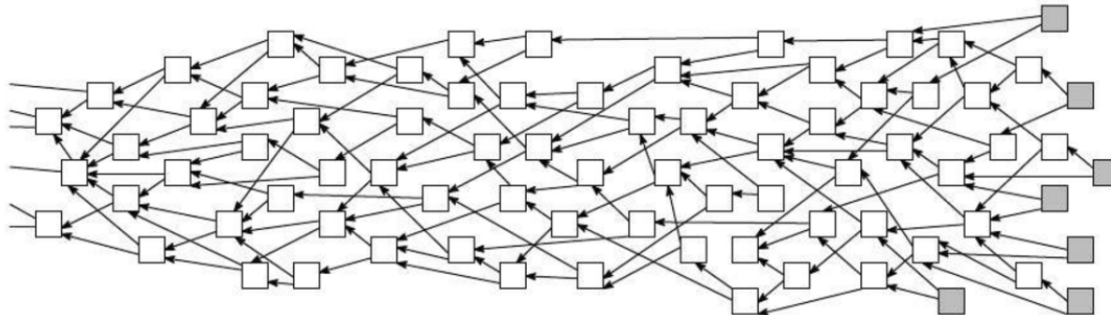
Blockchain



<https://en.wikipedia.org/wiki/Blockchain#/media/File:Blockchain.svg>

DAG

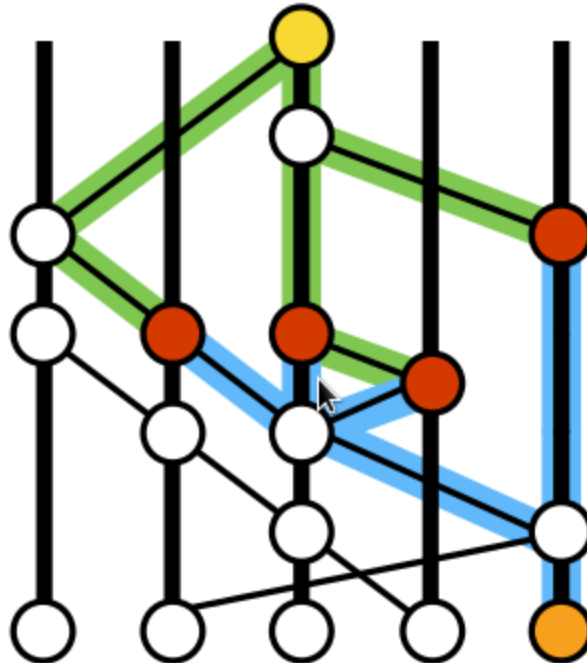
- Hashgraph, Avalanche, Tangle...
- Usually are not resistant to Sibyl attack (needs PoS or something else)



<https://ministryofblockchain.io/is-directed-acyclic-graph-dag-blockchains-new-competitor/>

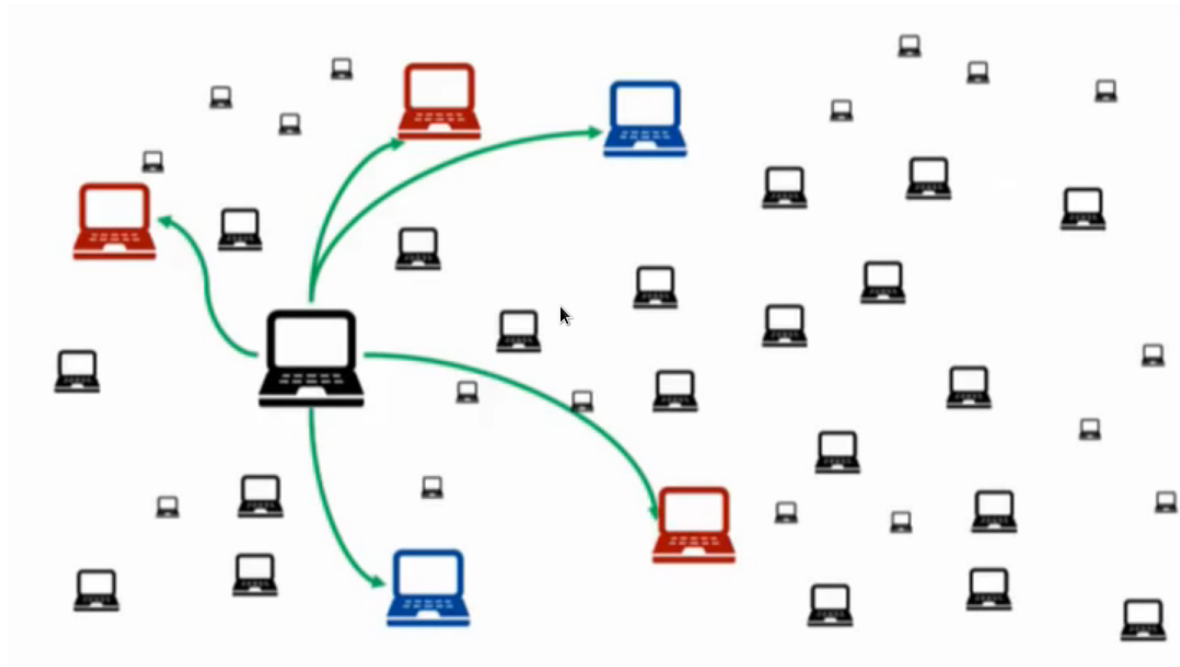
# HASHGRAPH

- Gossip about gossip
- Virtual voting



# AVALANCHE

- Gossip protocol
- Metastability



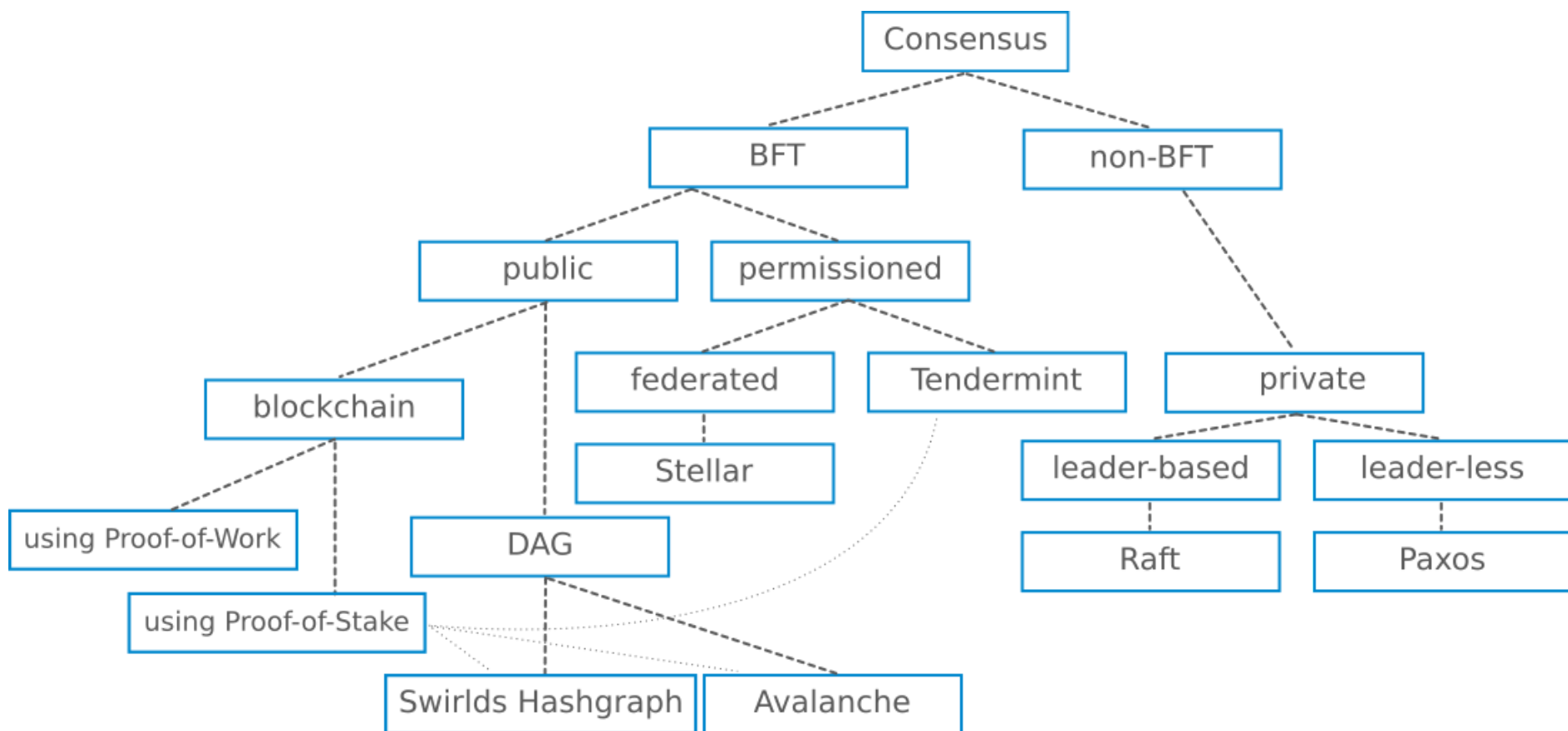
<https://www.youtube.com/watch?v=AXrrqtFIGow>

# DAG BASED CONSENSUS

## SUMMARY

- different data structure for storing transaction data
- not resistant to sibyls attack - PoS is usually involved

# CHALOUPKA-JURÁNEK TAXONOMY



# TAKE-OFFS

- Consensus protocol is a crucial part of any distributed ledger.
- Choice of consensus protocol influences heavily many characteristic of distributed ledger (including performance and security).
- There are several types of distributed ledgers, several families of consensus algorithms and not every consensus algorithm is suitable for every distributed ledger.

**QUESTIONS**

# LINKS

- [S. Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System](#)
- [D. Ongaro, J. Ousterhout, In Search of an Understandable Consensus Algorithm](#)
- [M. Castro, B Liskov, Practical Byzantine Fault Tolerance](#)
- [D. Mazieres, The Stellar Consensus Protocol](#)
- [The latest gossip on BFT consensus - Tendermint consensus algorithm](#)
- [L. Baird, The Swirlds Hashgraph Consensus Algorithm](#)
- [Team Rocket, Snowflake to Avalanche](#)
- [A Survey on Consensus Mechanisms and Mining Strategy Management in Blockchain Networks](#)

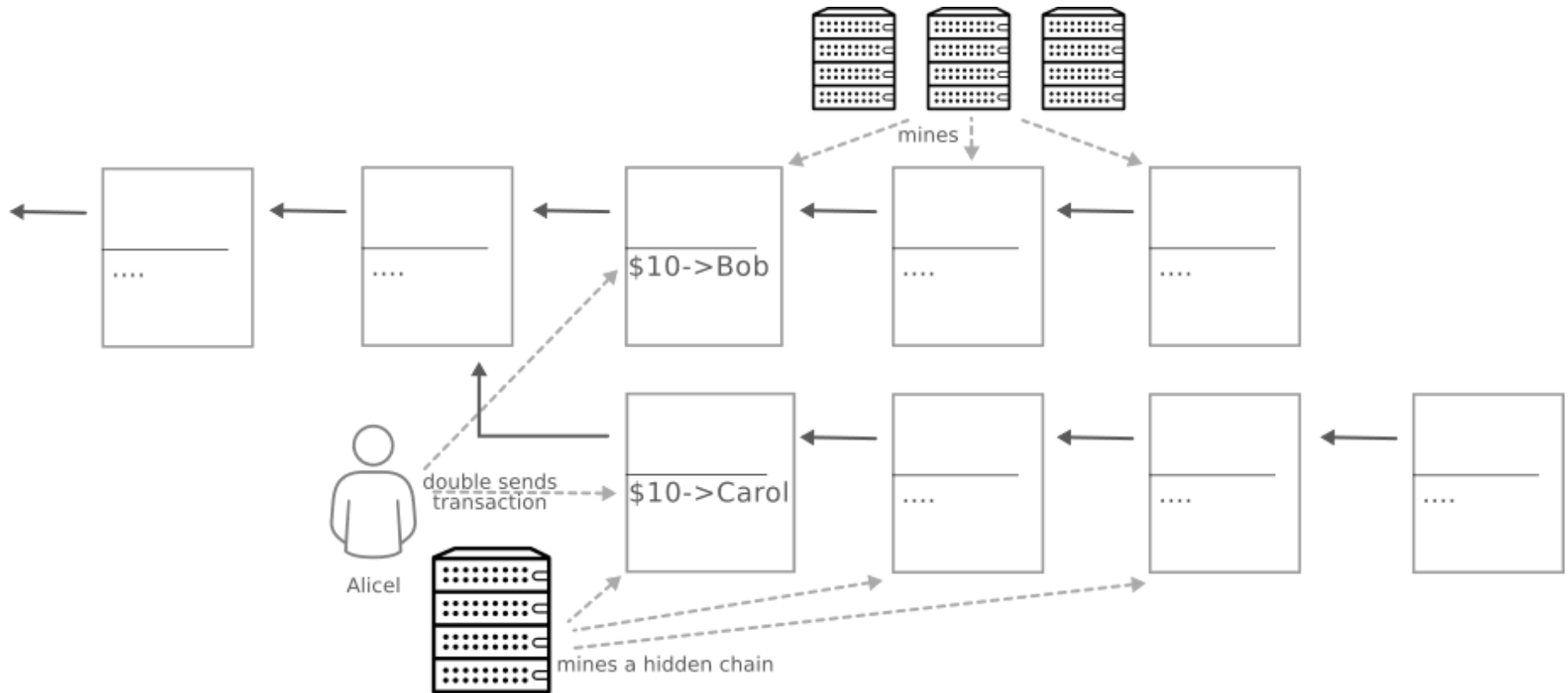




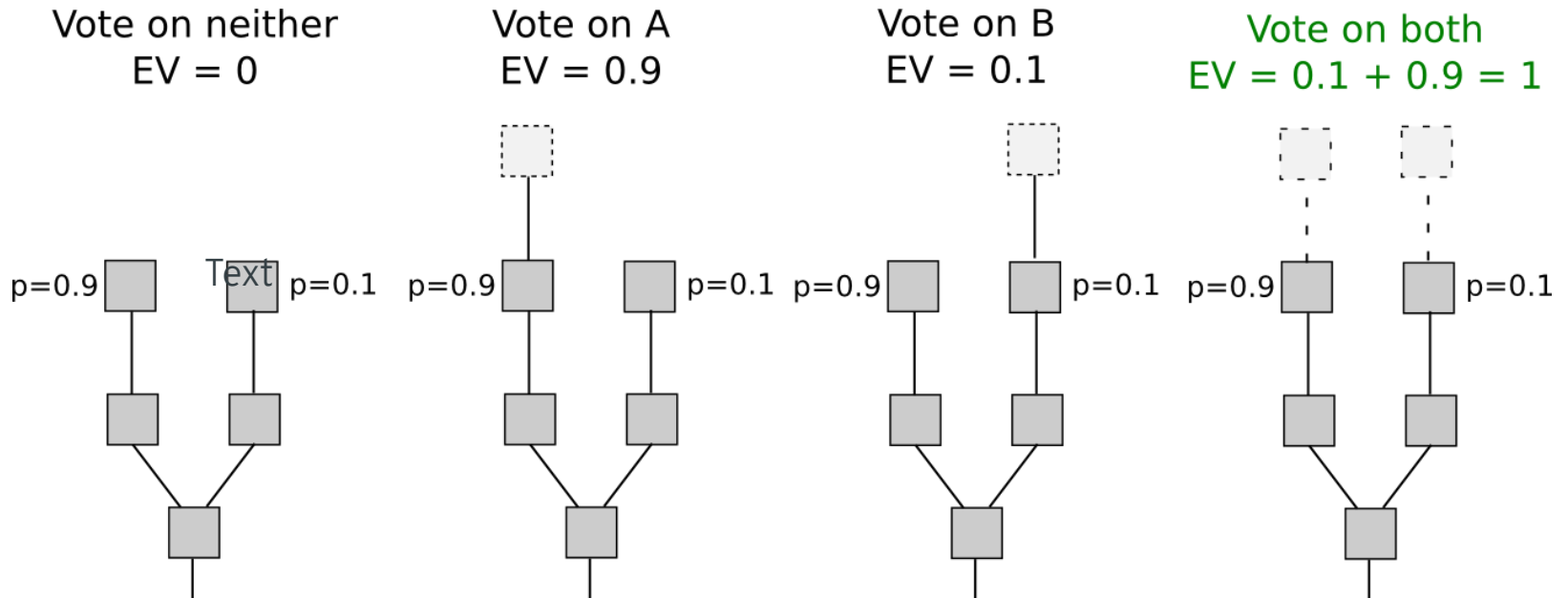
Thank you for your attention!

**BACKUP SLIDES**

# BITCOIN 51% ATTACK



# POS: NOTHING AT STAKE



# PROOF-OF-X

There are many:

- Proof of Capacity
- Proof of Elapsed Time (PoET)
- Proof of Authority
- Proof of Activity
- Proof of Burn
- Proof of Weight
- ...

Beware: Proof-of-X doesn't mean it's similar to PoW, actually in many cases it's quite different (e.g. centralized).

# TAXONOMY TABLE

Proof-of-Work	Bitcoin	Public Permissionless	Byzantine tolerant	Probabilistic
Proof-of-Stake	Ethereum 2.0	Public Permissionless	Byzantine tolerant	Finite
Delegated PoS	Stellar	Permissioned public	Byzantine tolerant	Finite
Raft	Hyperledger	Permissioned private	non-Byzantine tolerant	Finite
Tendermint	Cosmos	Permissionless public	Byzantine tolerant	Finite
Swirlds	Hedera Hashgraph	Permissionless public	Byzantine tolerant	Finite
Avalanche	Ava	Permissionless public	Byzantine tolerant	Probabilistic