Do We Need Consensus?

Roger Wattenhofer

ETH Zurich – Distributed Computing Group

Bitcoin: A Peer-to-Peer Electronic Cash System

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

"The problem of course is the payee can't verify that one of the owners did not double-spend the coin."

"We need a system for participants to agree on a single history of the order in which [transactions] were received."

no double-spending

≠

single order

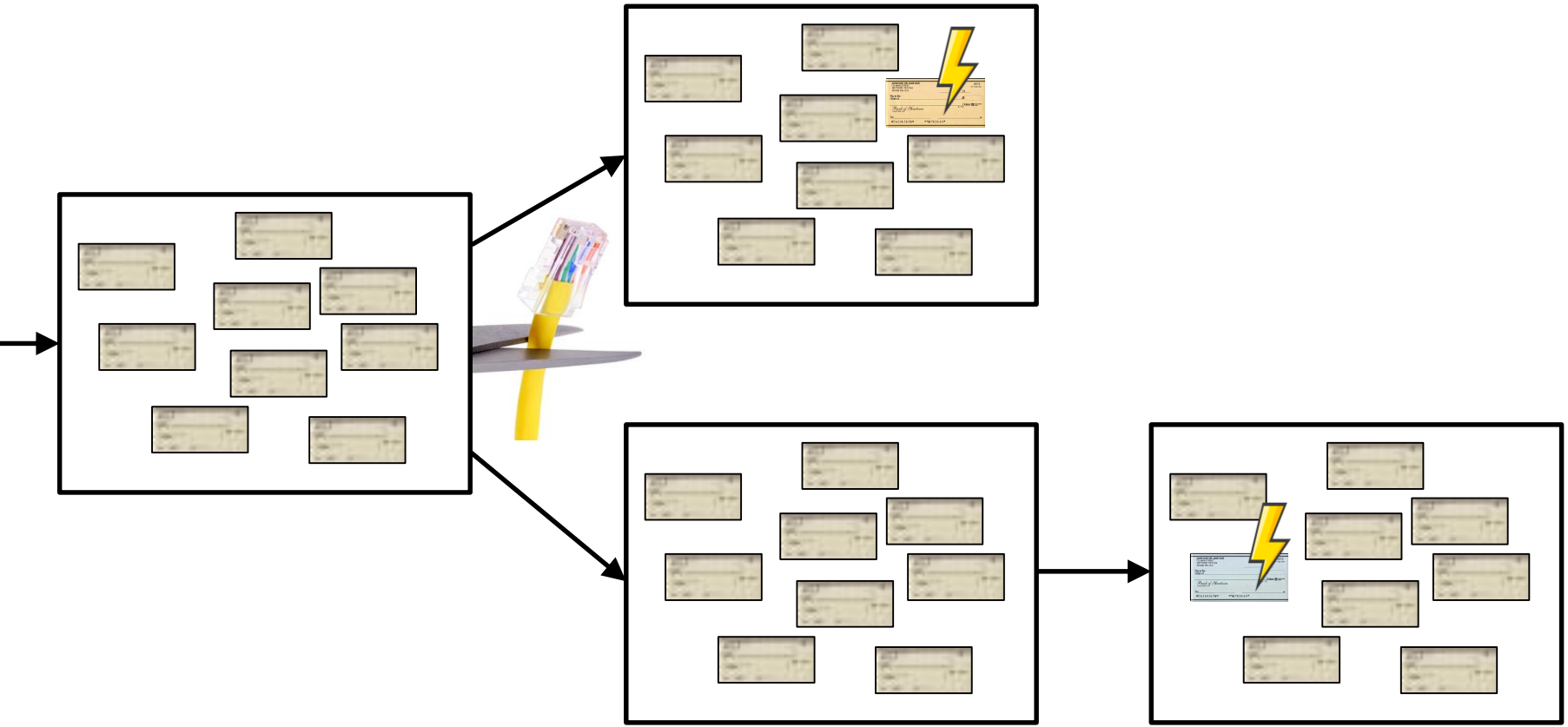
=

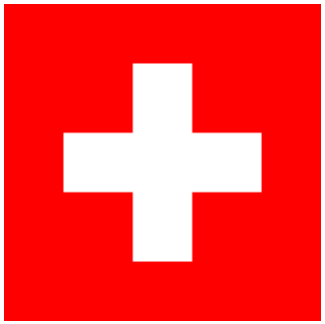
consensus

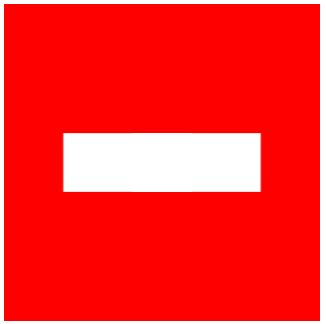
Double-Spending

Blockchains Solve Double-Spending Problem

What About Network Outages?







Unchangeable
Market Cap

Anonymous?
Permissionless?
Scalable = Secure?

Asynchrony
Finality
Throughput
Energy (PoW)
Smart Contracts
Unchangeable

Many Alternatives

	PBFT[1]	HoneyBadger BFT[10]	Broadcast-based[5]	Bitcoin and Ethereum[14]	Ouroboros[7]	Algorand[2]	ABC
Permissionless				✓	✓	✓	✓
Proof-of-work free	✓	✓	✓		✓	✓	✓
Finality	✓	✓	✓			✓	✓
Asynchronous		✓	✓				✓
Deterministic	✓		✓				✓
Parallelizable			✓				✓
General smart contracts	✓	✓		✓	✓	✓	

Without Consensus

A Non-Consensus Based Decentralized Financial Transaction Processing Model with Support for Efficient Auditing

by

Saurabh Gupta

A Thesis Presented in Partial Fulfi
of the Requirements for th
Master of

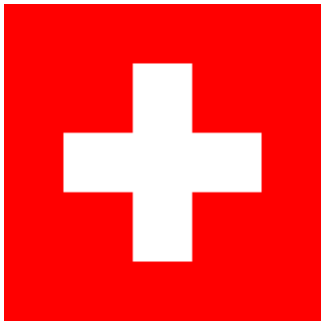
ABC: Asynchronous Blockchain without Consensus

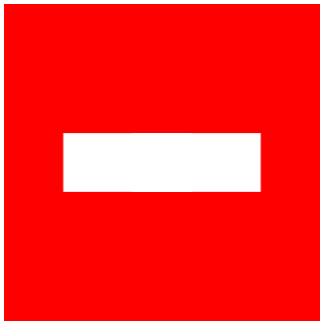
Jakub Sliwinski and Roger Wattenhofer

ETH Zurich

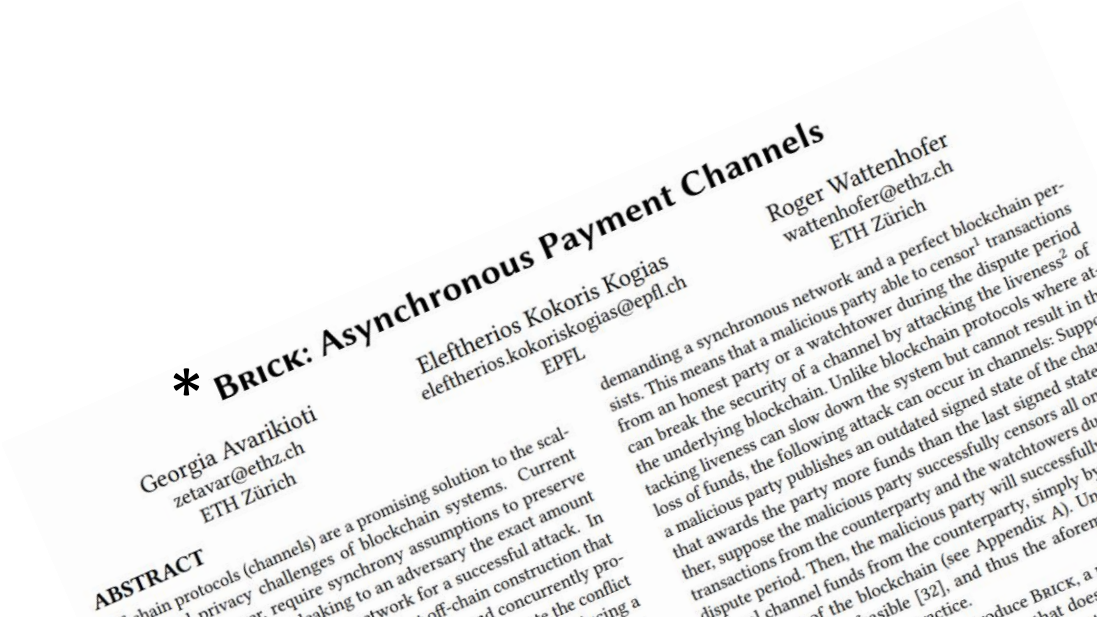
{jsliwinski,wattenhofer}@ethz.ch

conception that a blockchain needs c
ibuted property with a remar
consensus is at all ne
called ABC t
with an a





Asynchronous*
Throughput
Finality
Energy (PoS)
Permissionless
Scalable



* Brick: Asynchronous Payment Channels

Georgia Avarikioti
zetavar@ethz.ch
ETH Zürich

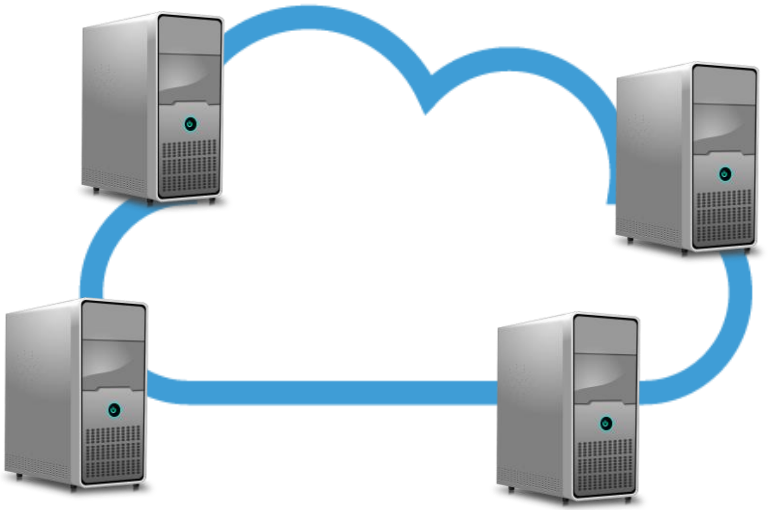
Eleftherios Kokoris Kogias
eleftherios.kokoriskogias@epfl.ch
EPFL

Roger Wattenhofer
wattenhofer@ethz.ch
ETH Zürich

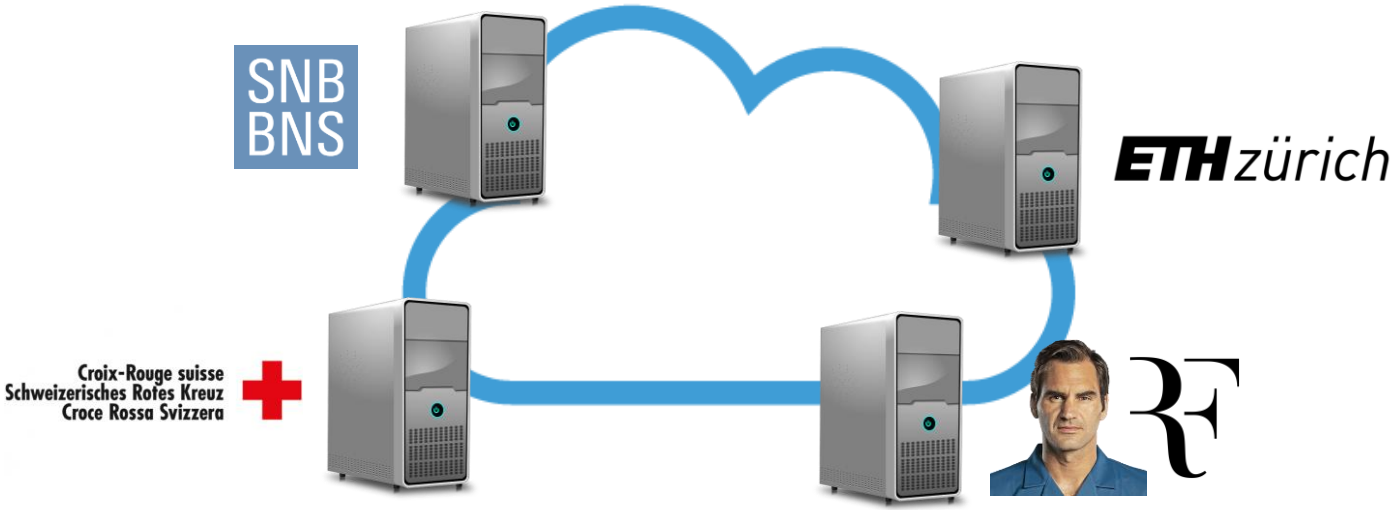
ABSTRACT

demanding a synchronous network and a perfect blockchain per-
sists. This means that a malicious party able to censor[1] transactions
from an honest party or a watchtower during the dispute period
can break the security of a channel by attacking the liveness[2] of
the underlying blockchain. Unlike blockchain protocols where at-
tacking liveness can slow down the system but cannot result in th
loss of funds, the following attack can occur in channels: Suppo
a malicious party publishes an outdated signed state of the cha
that awards the party more funds than the last signed state
ther, suppose the malicious party successfully censors all or
transactions from the counterparty and the watchtowers du
dispute period. Then, the malicious party will successfull
channel funds from the counterparty, simply by
of the blockchain (see Appendix A). Un
sible [32], and thus the aforen
ctice.
duce Brick, a
hat does

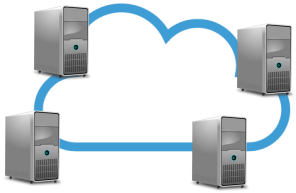
Permissioned ABC

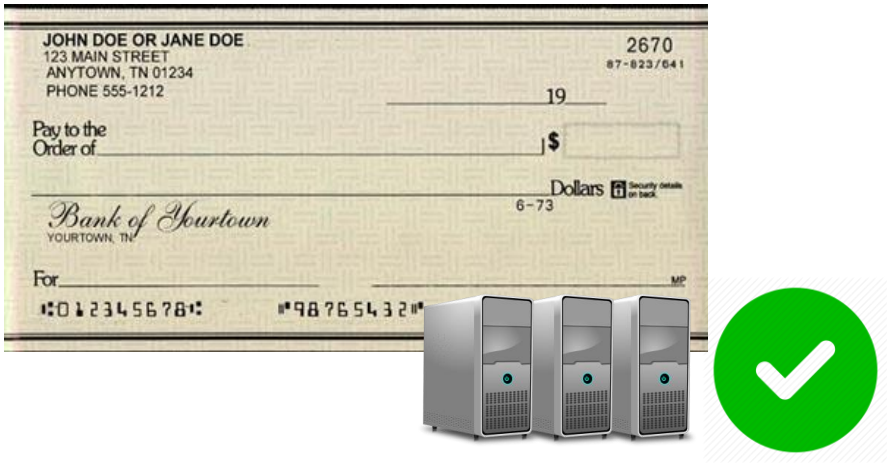


Permissioned ABC



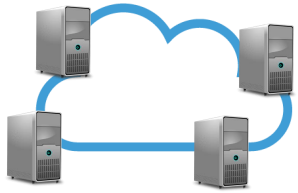
Permissioned ABC

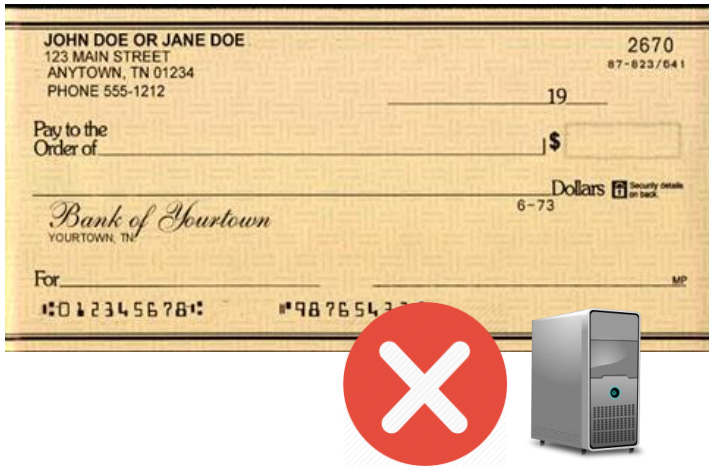


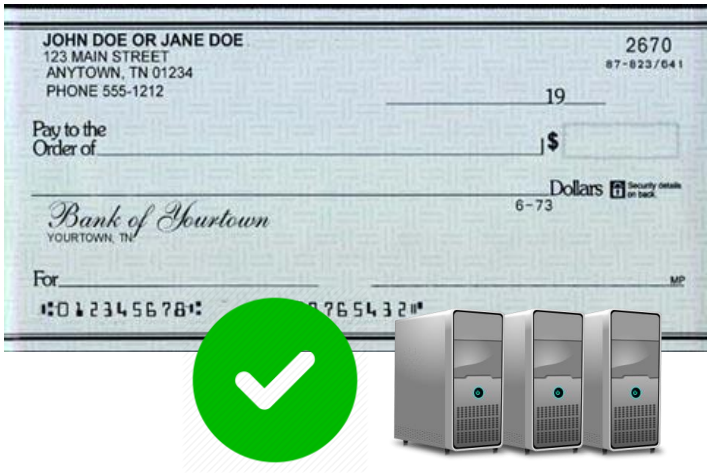


Needed: 3 out of 4 signatures

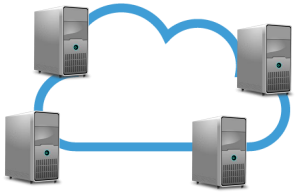
Double-Spending

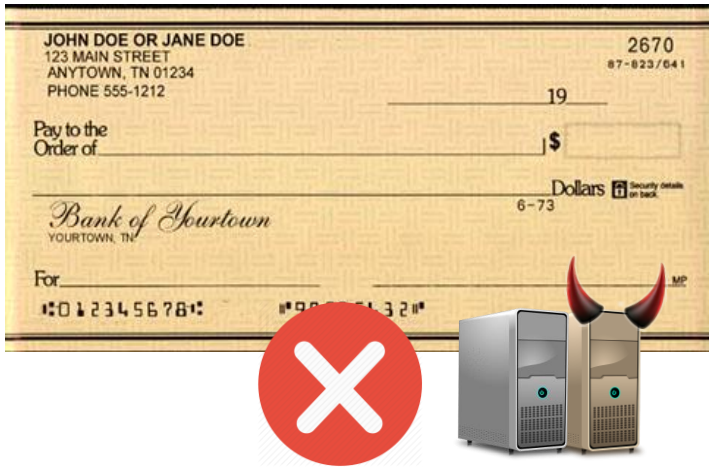


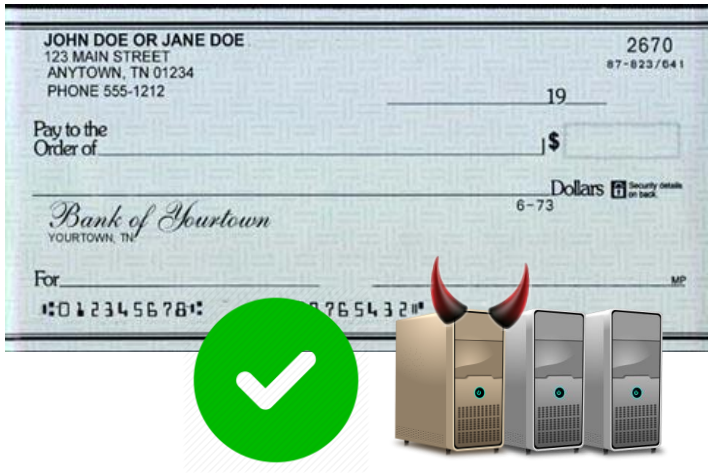




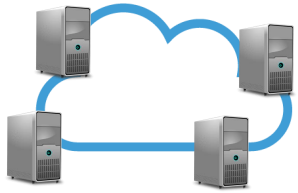
Double-Spending

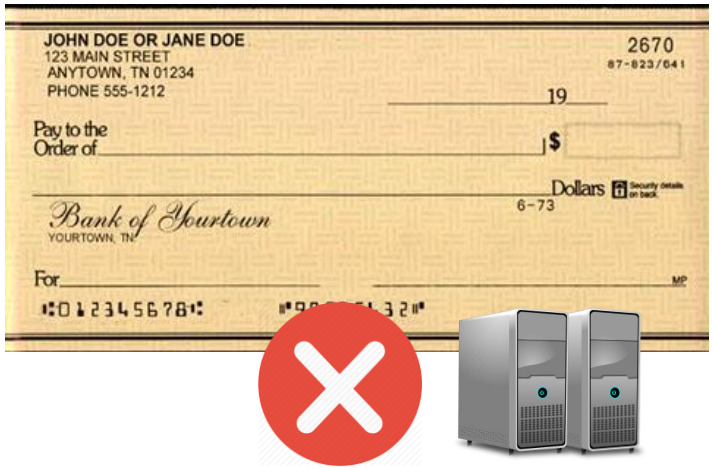


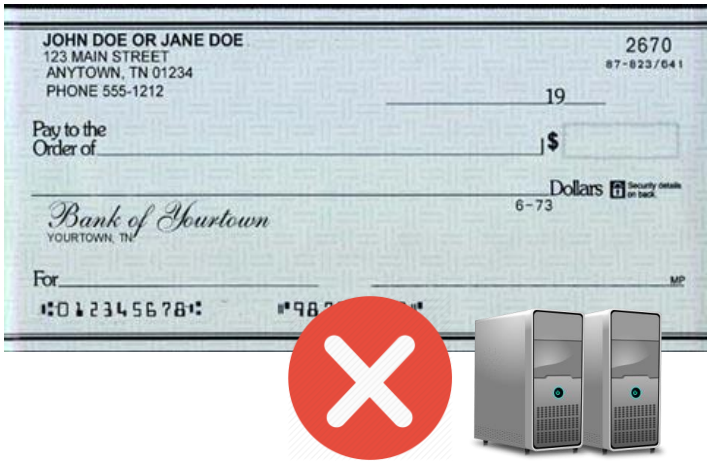




Double-Spending



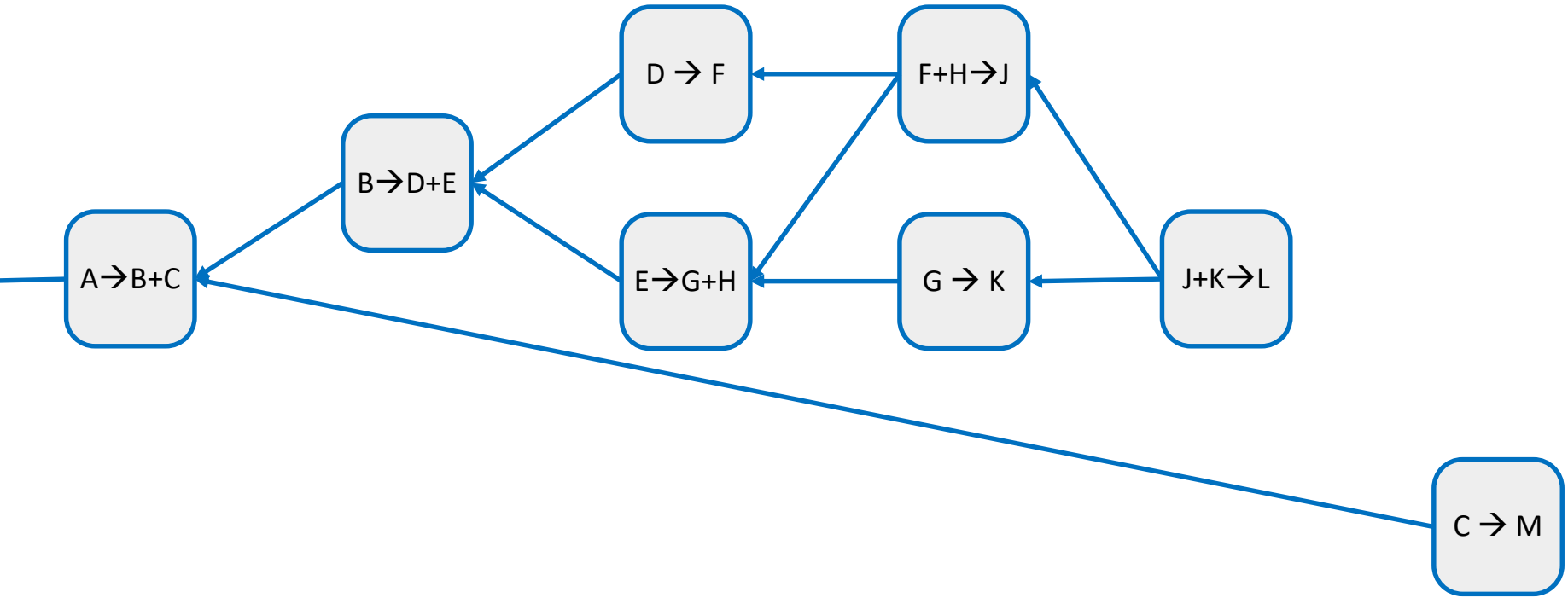




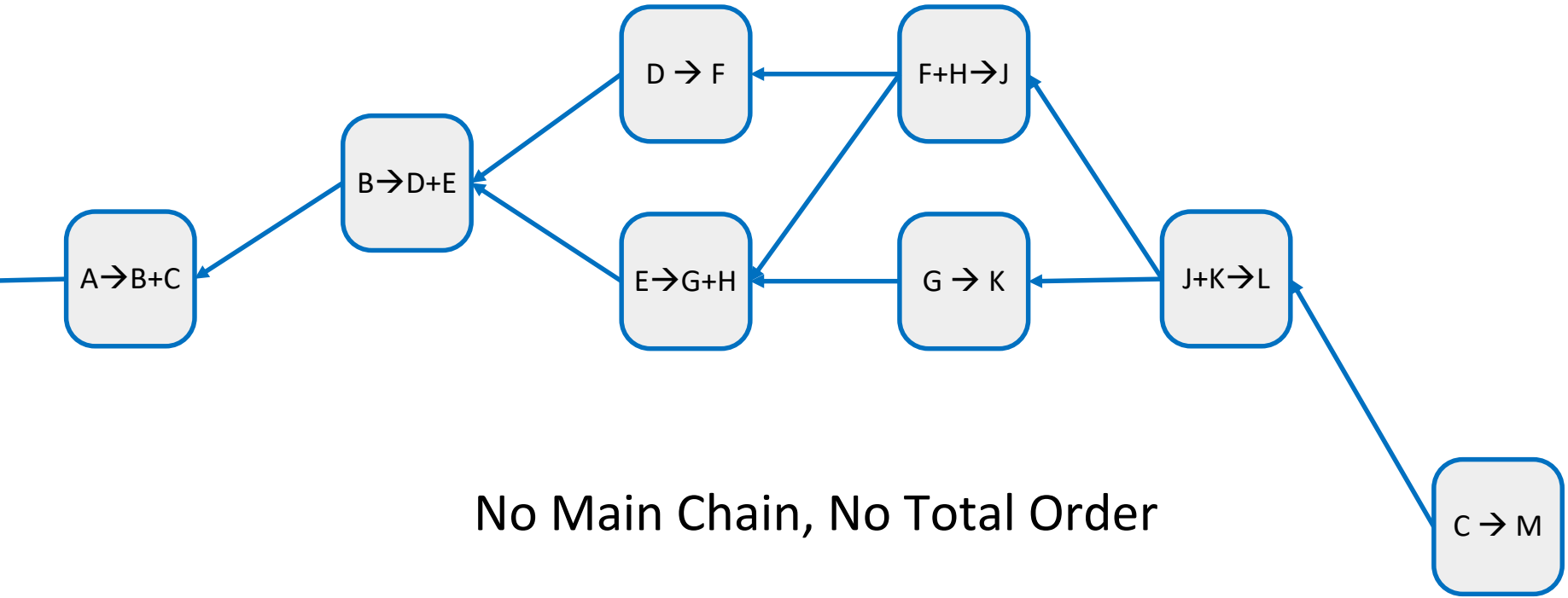
Usual Safety Condition

Less than 1/3 Byzantine

Point to Money Source

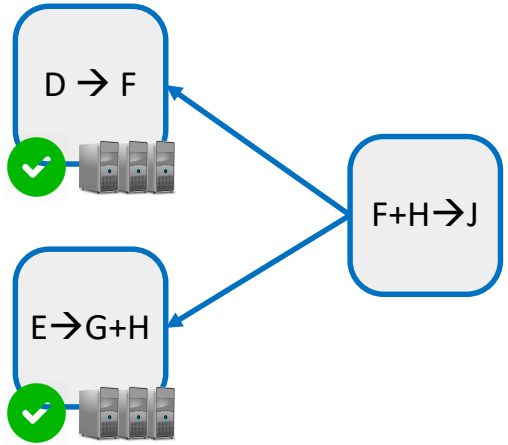


Point To All Transactions!

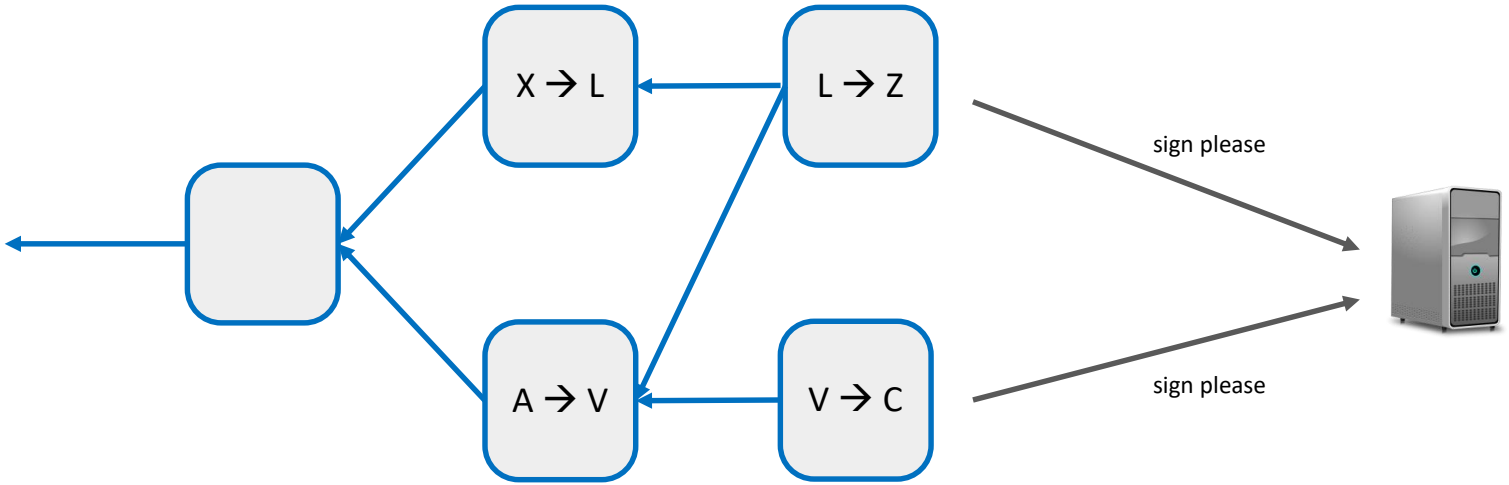


No Main Chain, No Total Order

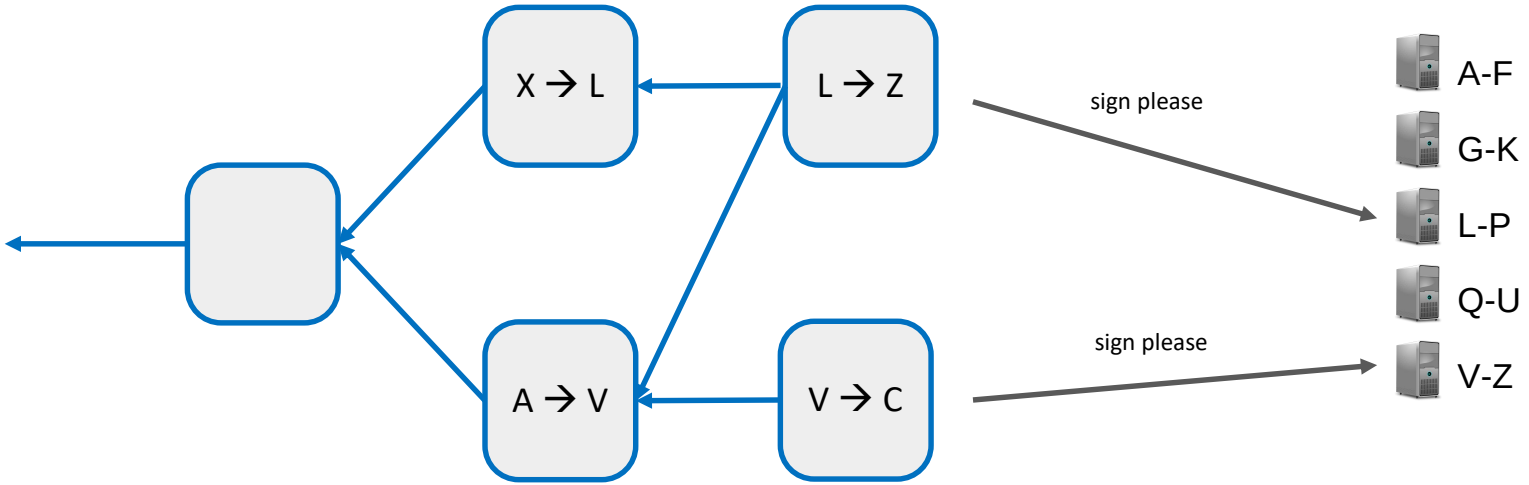
Asynchronous: Without Explicit DAG



Sharded Signing



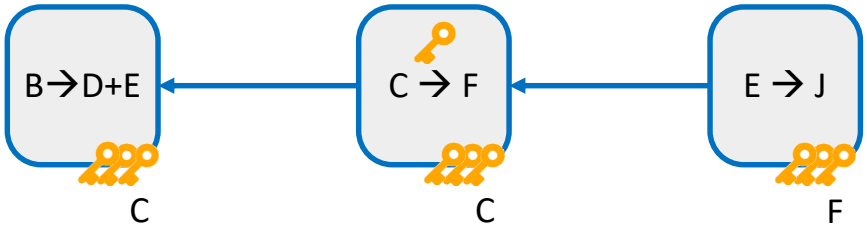
Sharded Signing



Also Permissionless?

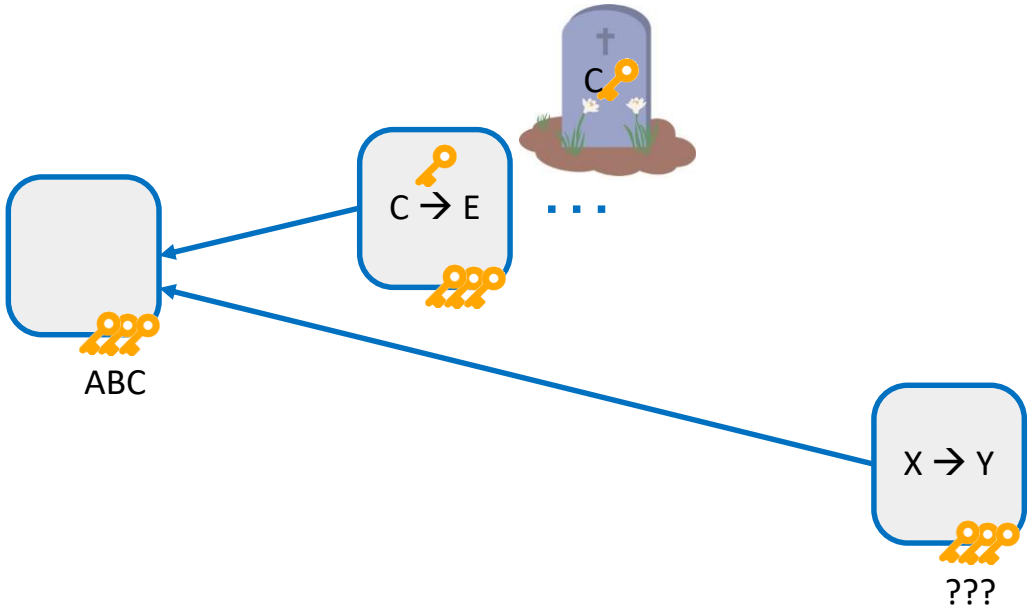
(Without Proof-of-Work)

1. Transferrable Signing Keys



2. Key Delegation (Pooling)

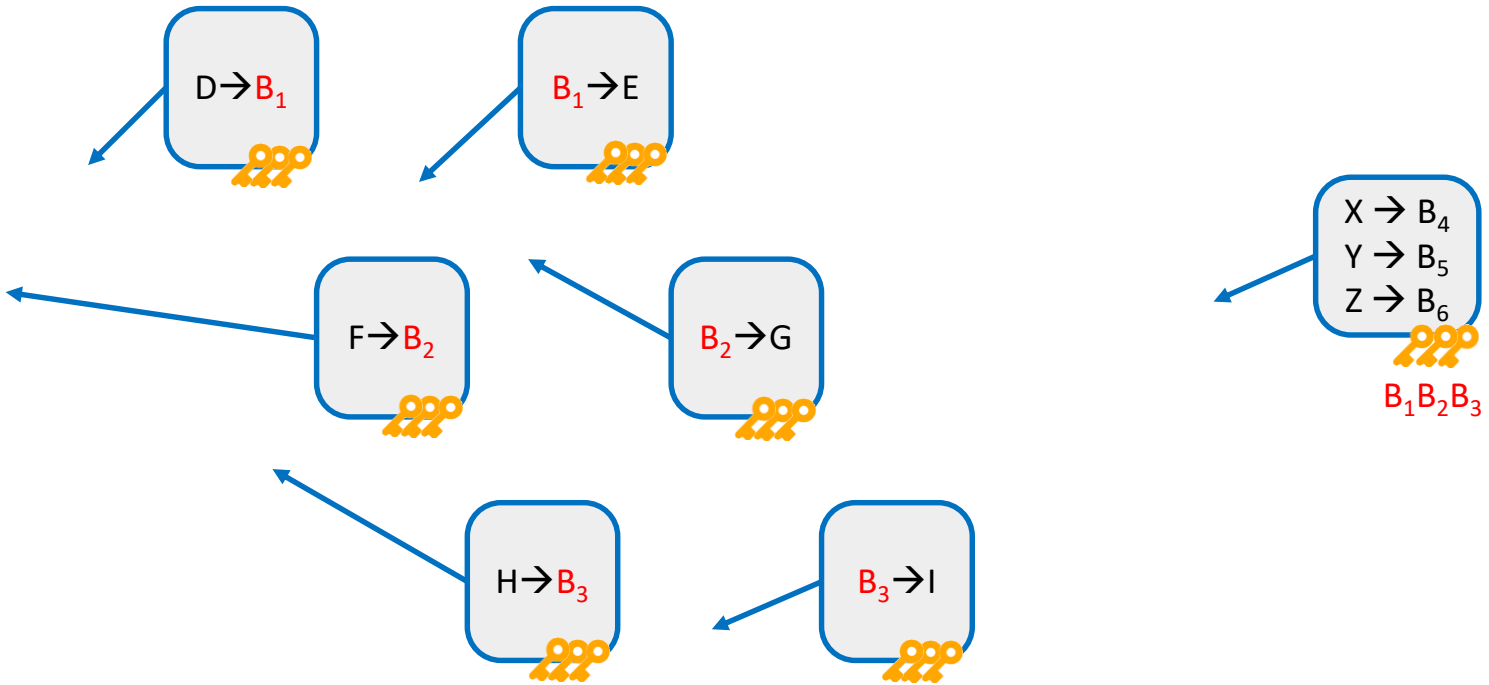
It’s Not So Easy



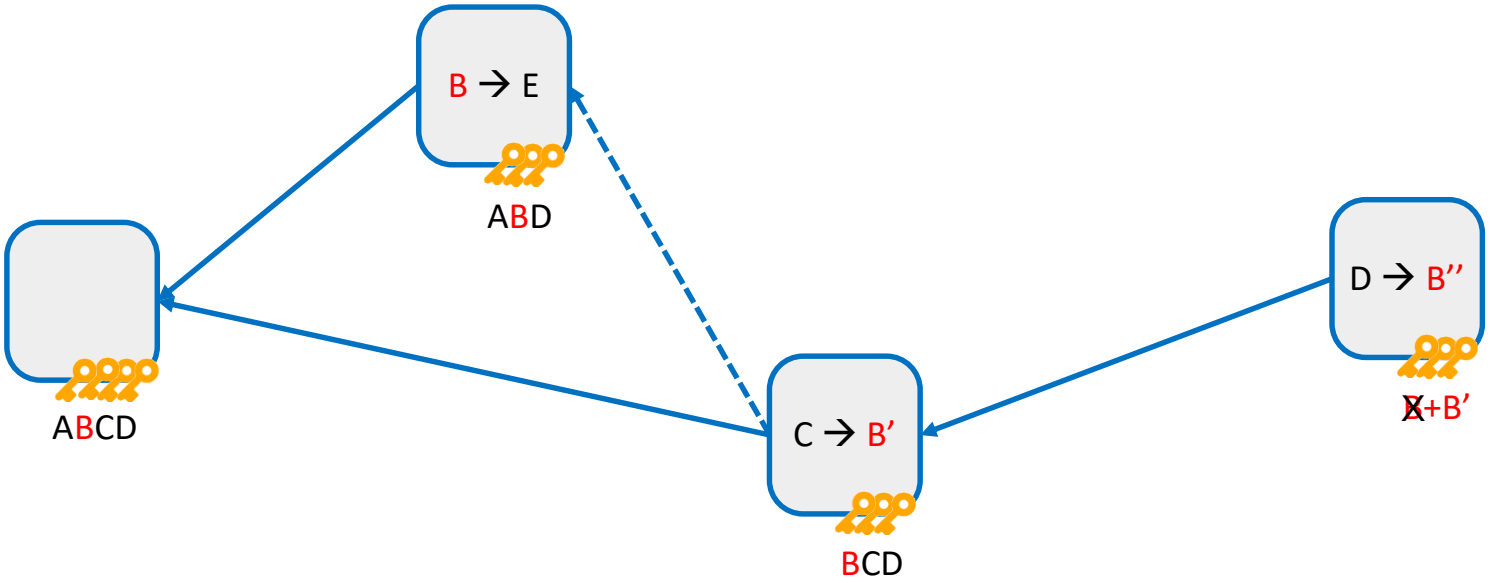
Usual Safety Condition

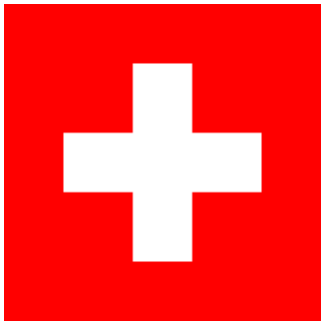
Byzantine $$$ Less Than 1/3 of Stake

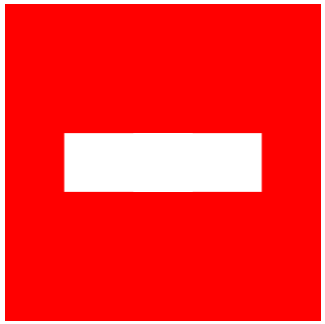
Byzantine Not Burying Keys…



Concrete Example







Asynchronous
Throughput
Finality
Energy (PoS)
Permissionless
Scalable

Smart Contracts?

Roger Wattenhofer, ETH Zurich, www.disco.ethz.ch


Ene, mene,
eins, zwei, drei,
Bitcoins bringe
mir herbei.
Hash Hash.
@grauhut
BiBi
BlockChain