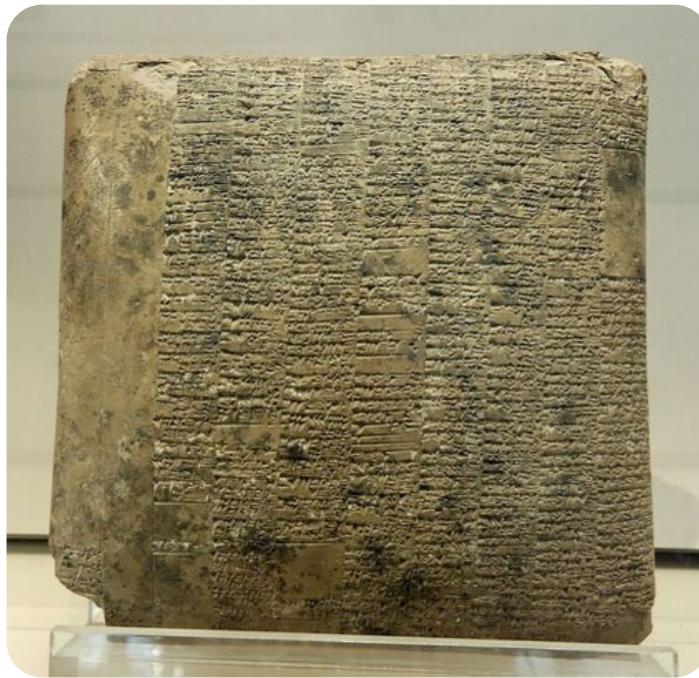


Building blocks

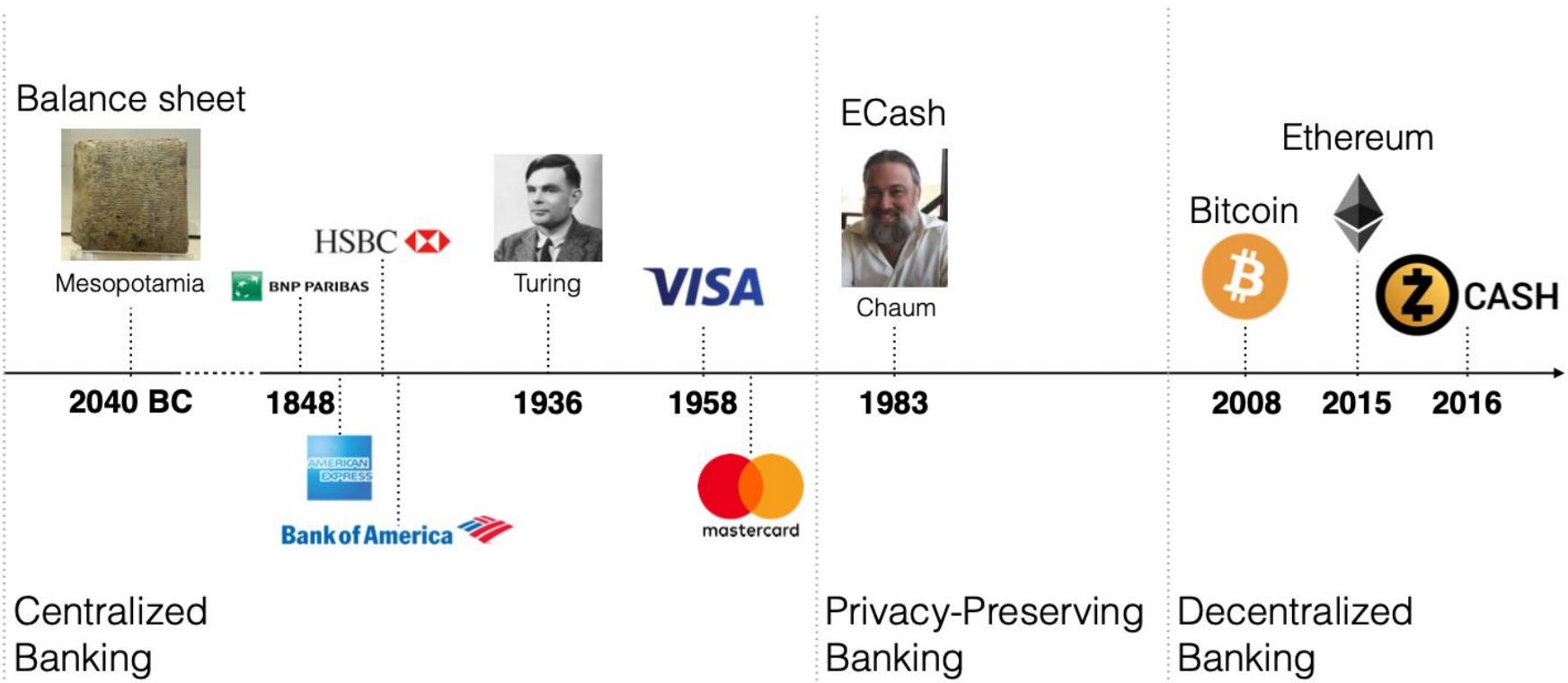
- 2040 BC



Balance sheet - Mesopotamia

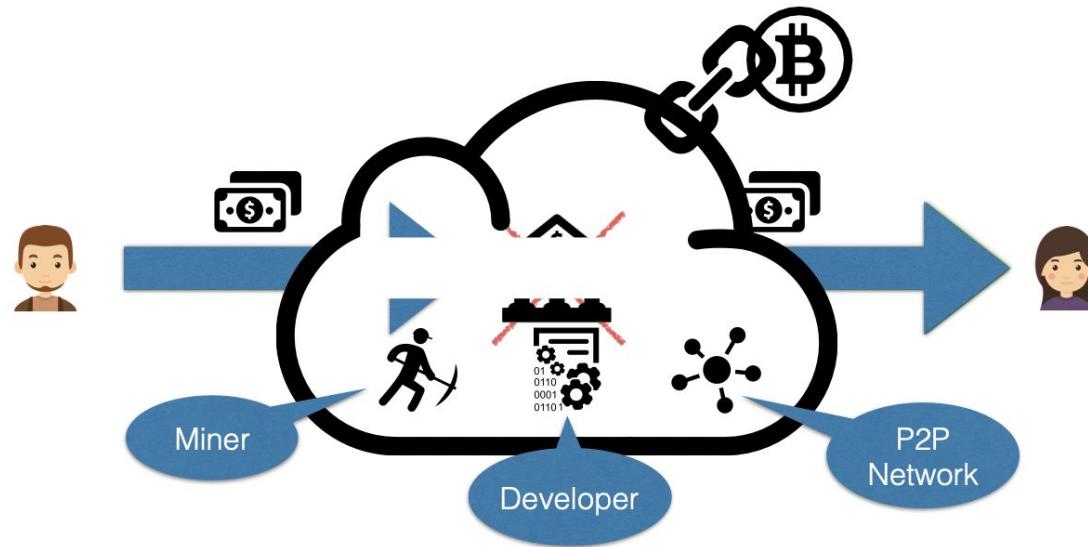
Building blocks

- Evolution of accounting



Building blocks

- From Centralized to Decentralized payment systems



- How to perform secure **decentralized** payments?
- How to exchange **privacy-preserving** payments?
- How to make decentralized systems **efficient**?

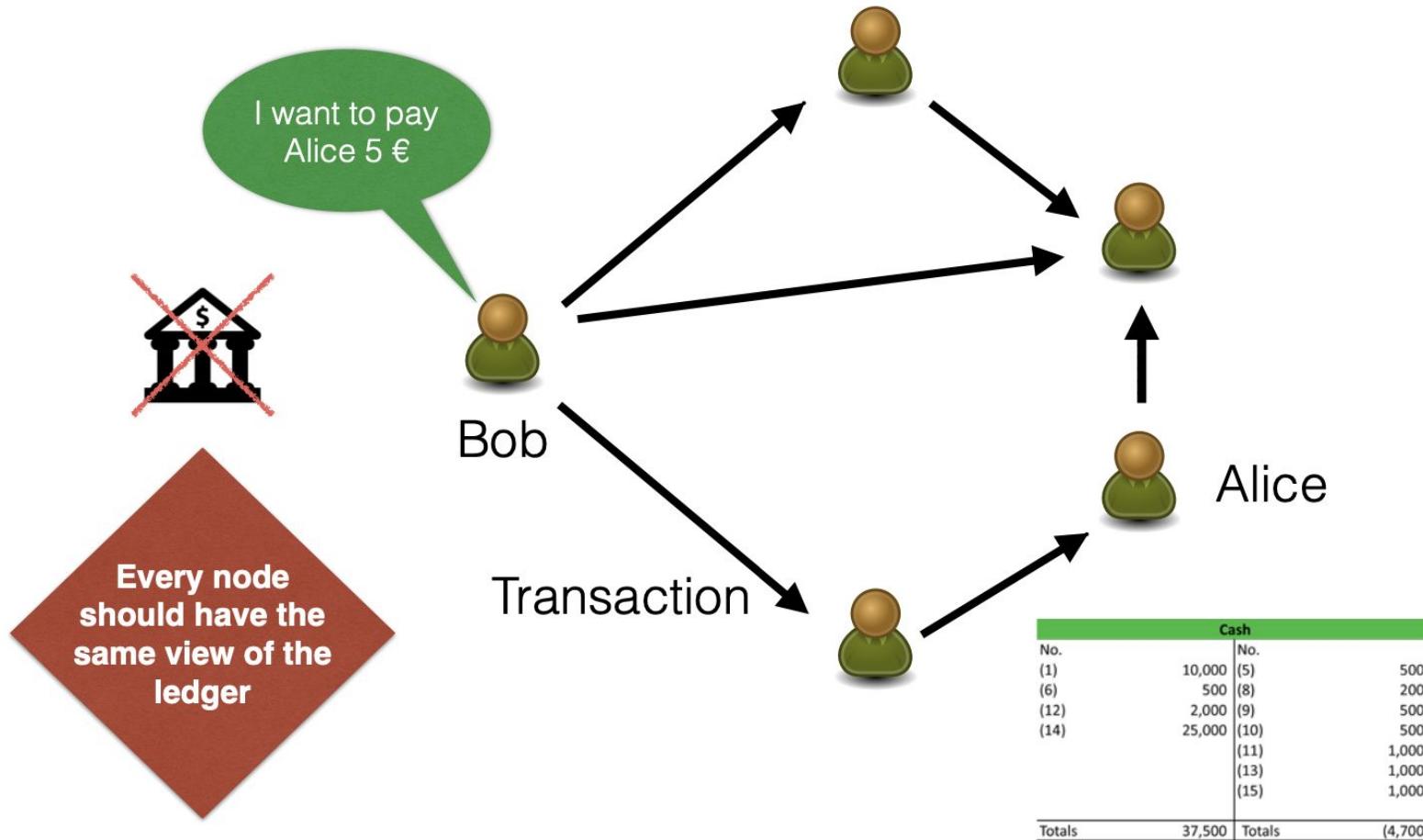
Building blocks

- Digital payment systems
 - With a centralized entity
 - Banks
 - Digital signatures are main double-spending resistance mechanism
 - Examples
 - Pepper Micropayments [Rivest]
 - ECash [David Chaum]
 - Privacy preserving



Building blocks

- Decentralized digital payments



Building blocks

- A Blockchain is a (distributed) database



Building blocks

- Blockchain participants
 - Writer

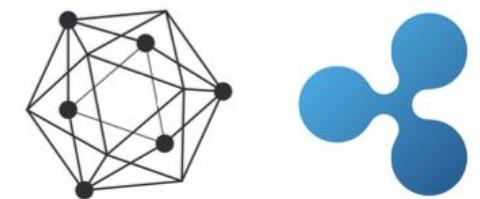


- Reader



Building blocks

- Blockchain != Blockchain
 - “Open and Decentralised” Blockchains
 - Bitcoin
 - Ethereum
 - ...
 - Permission-based Blockchains
 - Hyperledger
 - Ripple
 - Stellar
 - ...



Building blocks

- “Open and Decentralised” Blockchains



Building blocks

- Bitcoin
 - First introduced in 2009
 - By a pseudonym Satoshi Nakamoto
 - Peer-to-peer decentralized currency
 - No trusted third parties
 - It's Blockchain is Distributed DB
 - Transactions
 - Blocks
 - $0 < \text{Balance} < 21 \text{ Million Bitcoin}$
 - $1 \text{ Satoshi} = 10^{-8} \text{ Bitcoin}$



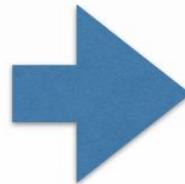
Building blocks

- Data types
 - Cryptographic Hash Functions



- SHA256
- RIPEMD160

Arbitrarily long
data

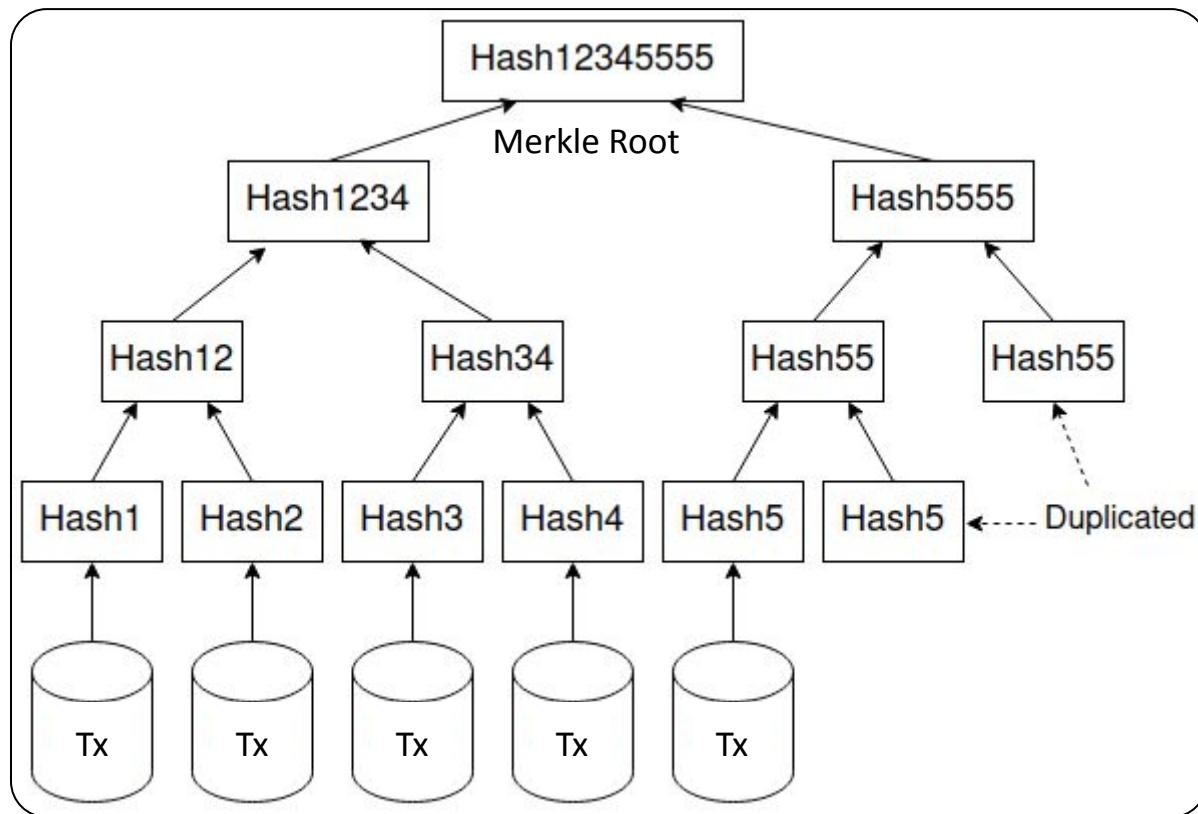


Fixed sized
hash/digest



Building blocks

- Data types
 - Merkle trees



Building blocks

- Data types
 - Elliptic Curve Signature Algorithm (ECDSA)



- ECDSA (secp256k1 curve) is used to
 - Sign transactions
 - Verify the signature of transactions
- Nothing in Bitcoin is encrypted

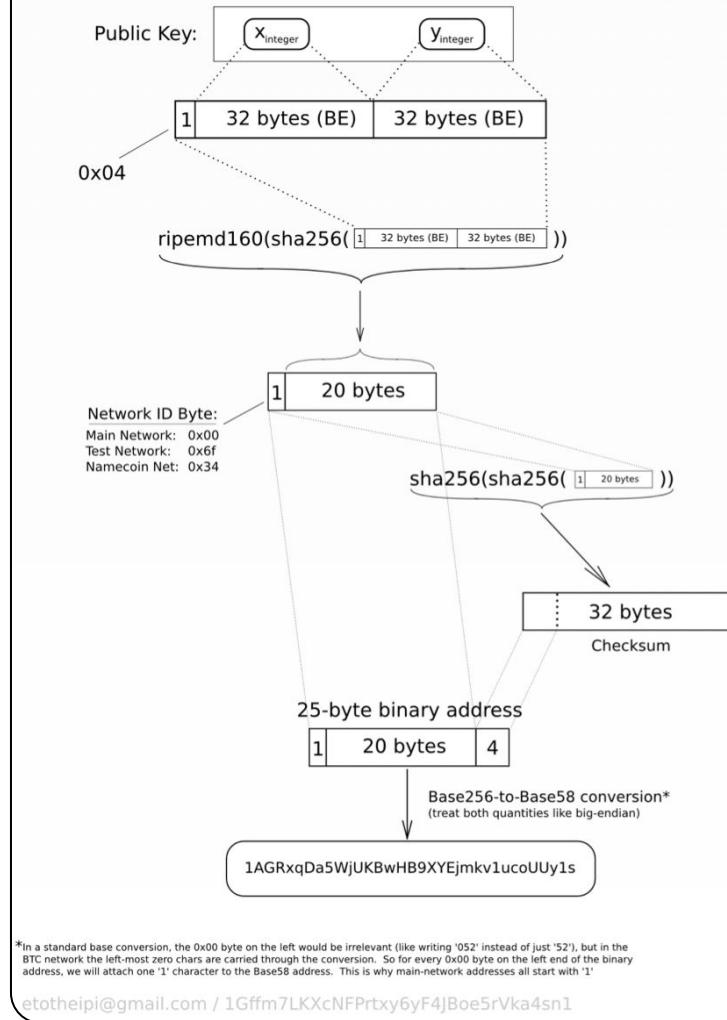


Building blocks

- Addresses
 - Unique identifier
 - Hash of a public key

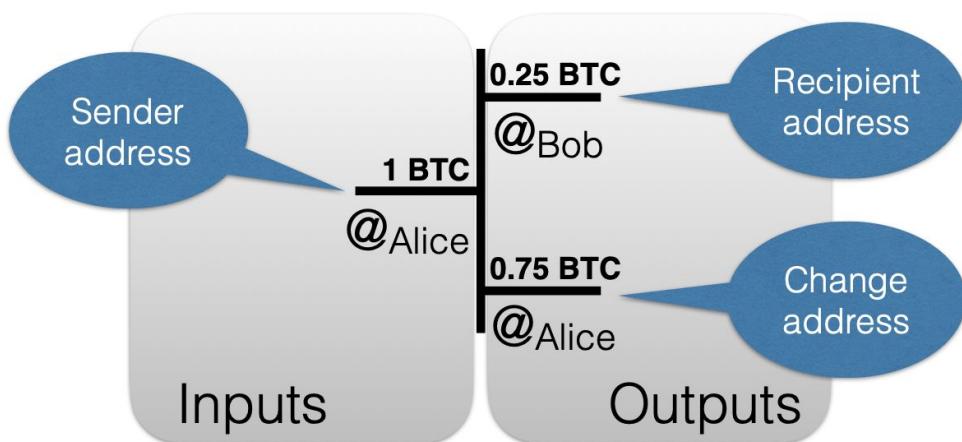
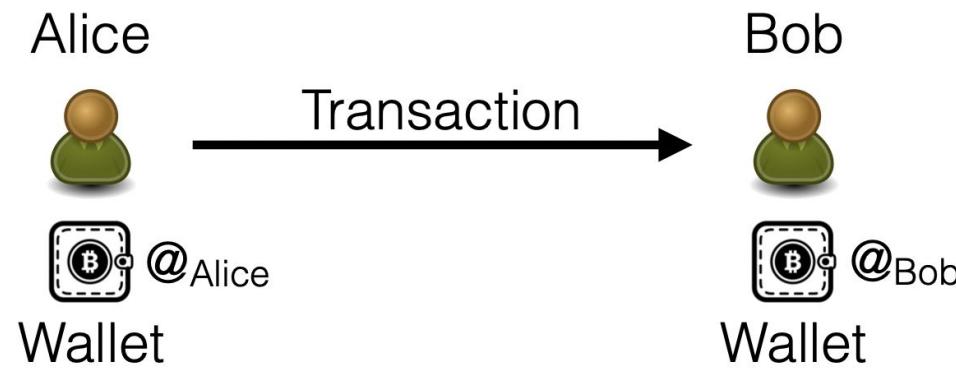
Type	Version prefix (hex)	Base-58 result's prefix
P2PKH address	0x00	1
P2SH Address	0x05	3
Bitcoin Testnet Address	0x6F	m or n
Private Key WIF	0x80	5, K, or L
BIP32 Extended Public Key	0x0488B21E	xpub
Bech32 (SegWit)	0x0303000203 (for checksum only)	bc1

Elliptic-Curve Public Key to BTC Address conversion



Building blocks

- Transactions



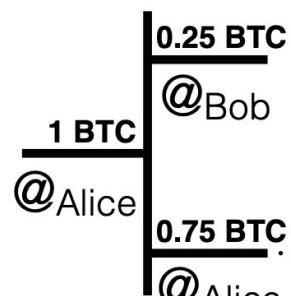
Transaction Fees

$$\sum \text{inputs} \geq \sum \text{outputs}$$

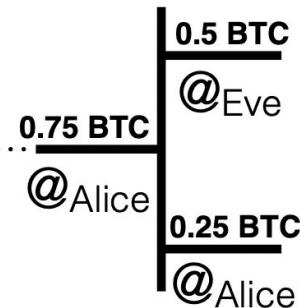
Difference are fees

Building blocks

- Transactions



Transaction 1



Transaction 2

Building blocks

- Script
 - Stack based programming language
 - Turing incomplete
 - If evals to *true* → Bitcoin transaction is valid
 - Many opcodes
 - Execution time is critical to prevent DoS attacks
 - E.g.,
 - <signature><publicKey>
 - OP_CHECKSIG



Constants are pushed onto the stack

Operation executes on stack values

Building blocks

- Transactions in Ethereum
 - Transaction with signature which modifies an **account**



type	from	sig	nonce	to	data	value	gaslimit	gasprice
------	------	-----	-------	----	------	-------	----------	----------

Building blocks

- Ethereum Virtual Machine
 - EVM Features
 - Stack of max depth of 1024
 - 32-byte words
 - Dedicated crypto opcodes
 - SHA-3
 - Big num multiply
 - GF-256 operators



EVM code

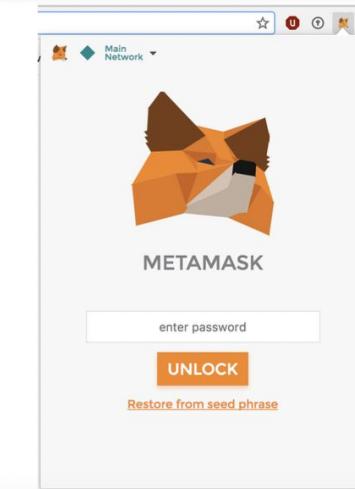
```
.code
PUSH 60           contract Ballot {\n
    struct...
PUSH 40           contract Ballot {\n
    struct...
MSTORE            contract Ballot {\n
    struct...
CALLVALUE         function Ballot(uint8 _numProp...
ISZERO             function Ballot(uint8 _numProp...
PUSH [tag] 1       function Ballot(uint8 _numProp...
JUMPI              function Ballot(uint8 _numProp...
PUSH 0              function Ballot(uint8 _numProp...
DUP1                function Ballot(uint8 _numProp...
REVERT              function Ballot(uint8 _numProp...
tag 1               function Ballot(uint8 _numProp...
JUMPDEST           function Ballot(uint8 _numProp...
PUSH 40             function Ballot(uint8 _numProp...
MLOAD              function Ballot(uint8 _numProp...
PUSH 20             function Ballot(uint8 _numProp...
DUP1                function Ballot(uint8 _numProp...
PUSHSIZE            function Ballot(uint8 _numProp...
DUP4                function Ballot(uint8 _numProp...
CODECOPY           function Ballot(uint8 _numProp...
DUP2                function Ballot(uint8 _numProp...
ADD                 function Ballot(uint8 _numProp...
PUSH 40             function Ballot(uint8 _numProp...
MSTORE            function Ballot(uint8 _numProp...
DUP1                function Ballot(uint8 _numProp...
DUP1                function Ballot(uint8 _numProp...
```

Building blocks

- Wallets
 - Bitcoin Core
 - Full node, downloads all txs
 - Bitcoin Wallet
 - Android
 - Jaxx
 - Multi-chain wallet
 - Ledger Nano S



The MyEtherWallet.com interface includes a "Create New Wallet" form where users can enter a password to encrypt their private key. It also features a "How to Create a Wallet" guide with step-by-step instructions and links to various wallet-related resources.

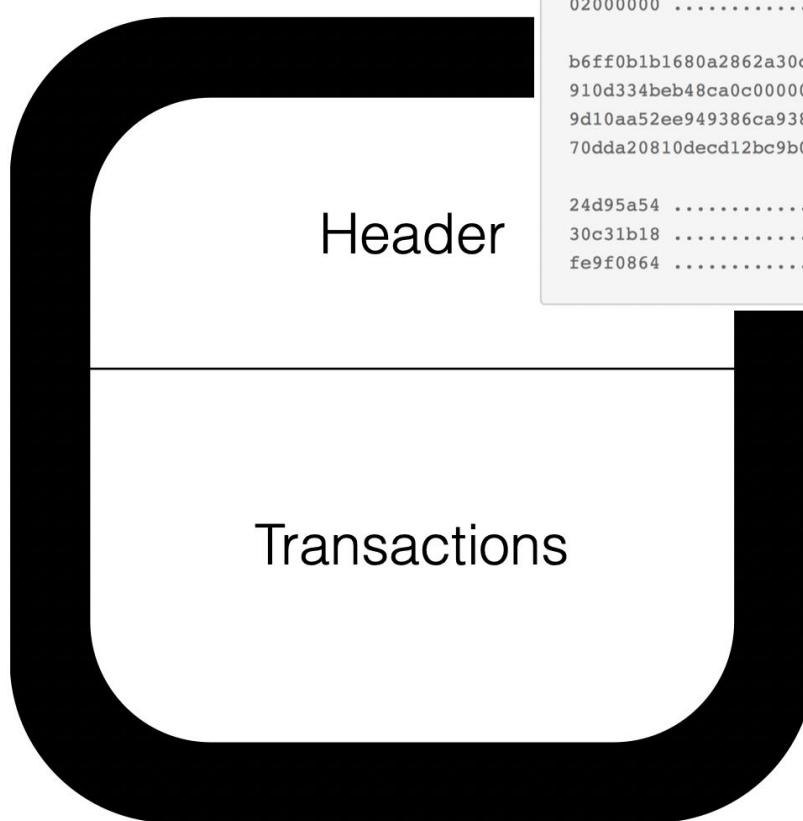


Building blocks

- What can you do with Crypto?
 - Trade with other people
 - Exchange to other crypto/FIAT
 - Buy something online
 - Lend
 - Donate
 - Build applications/games
 - ...

Building blocks

- Block

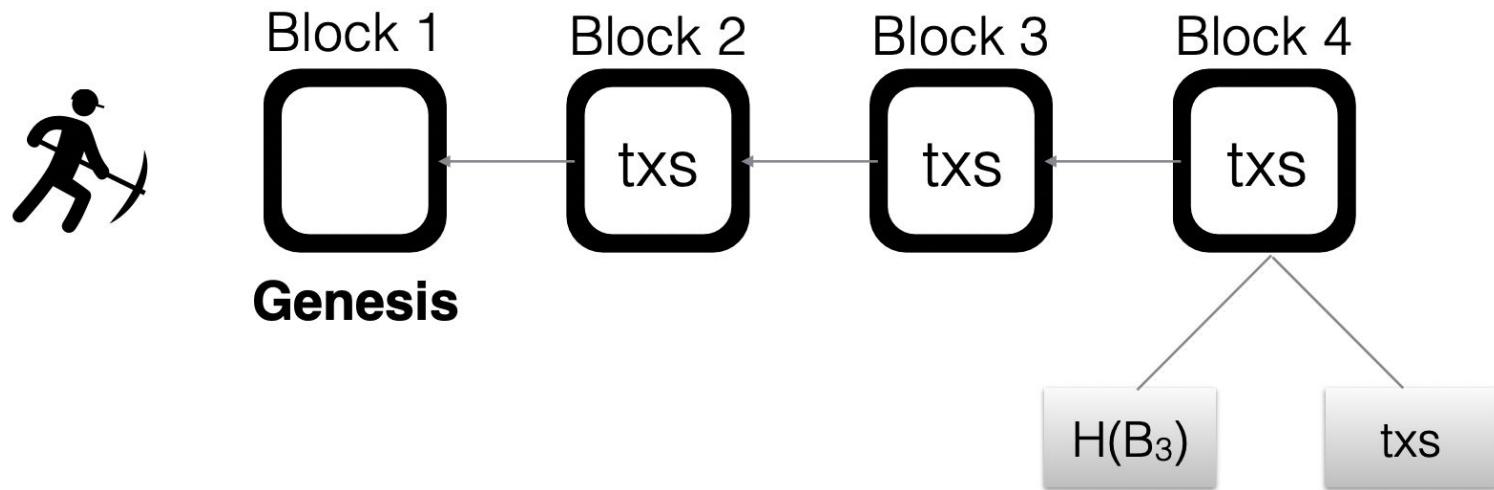


```
02000000 ..... Block version: 2
b6ff0b1b1680a2862a30ca44d346d9e8
910d334beb48ca0c0000000000000000 ... Hash of previous block's header
9d10aa52ee949386ca9385695f04ede2
70dda20810decd12bc9b048aab31471 ... Merkle root

24d95a54 ..... Unix time: 1415239972
30c31b18 ..... Target: 0x1bc330 * 256** (0x18-3)
fe9f0864 ..... Nonce
```

Building blocks

- Blockchain



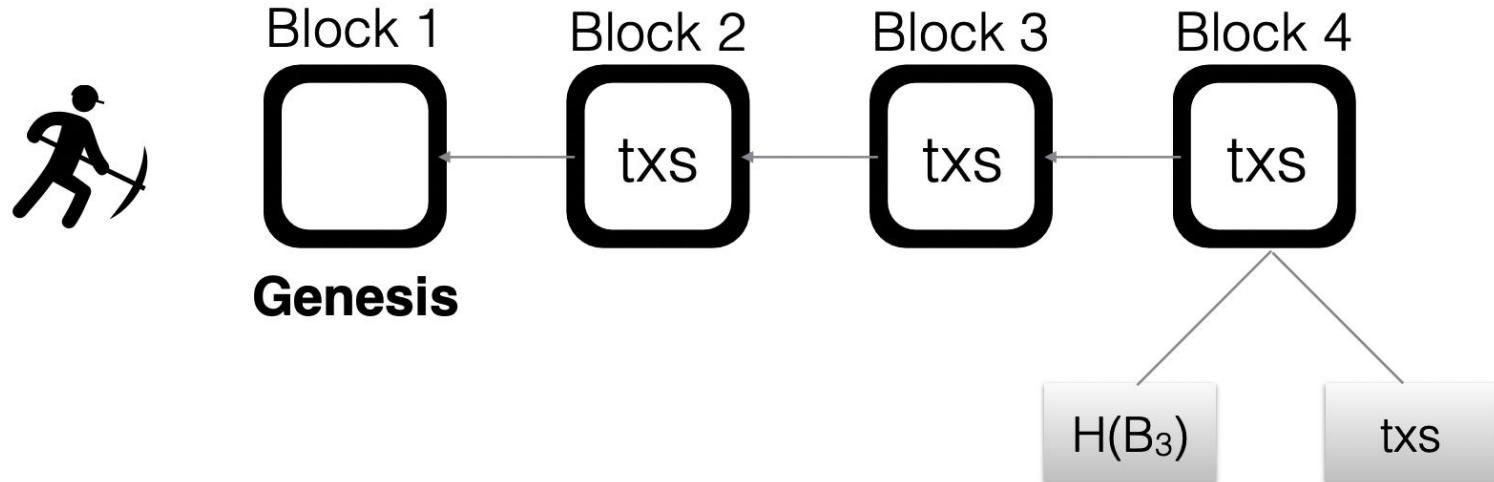
Building blocks

- Mining
 - Join the network, listen for transactions
 - Validate all incoming transactions
 - Listen for new blocks, build on valid blocks
 - Construct a block template
 - Find a nonce that validates the new block
 - Tell all the other miners
 - Receive reward



Building blocks

- Blockchain



- Mining

- Find Nonce N, such that

$\text{Hash}(\text{Hash}(B_3)|\text{txs}|N) < \text{target}$

Best known approach: **Brute Force**

Controls the **difficult**

Building blocks

- Mining difficulty

$\text{Hash}(\text{Hash}(B_3) | \text{txs} | N) < \text{target} = 0x\underline{000}^{**}$

~~$\text{Hash}(\text{Block_3} | \text{merkle_root} | 0xabca) = 0x03ef..$~~

~~$\text{Hash}(\text{Block_3} | \text{merkle_root} | 0xabcb) = 0x12ef..$~~

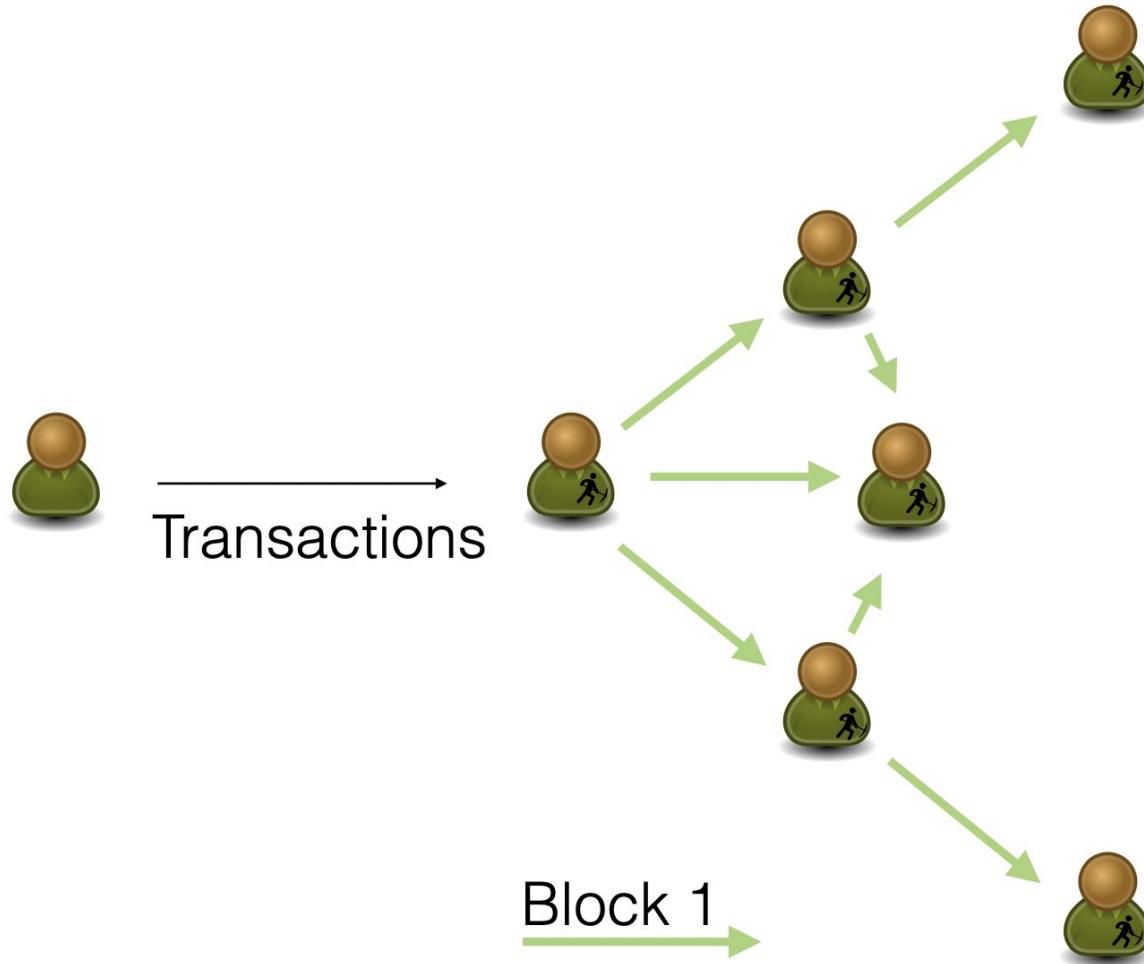
~~$\text{Hash}(\text{Block_3} | \text{merkle_root} | 0xabcc) = 0x20ef..$~~

$\text{Hash}(\text{Block_3} | \text{merkle_root} | 0xabcd) = 0x000f..$



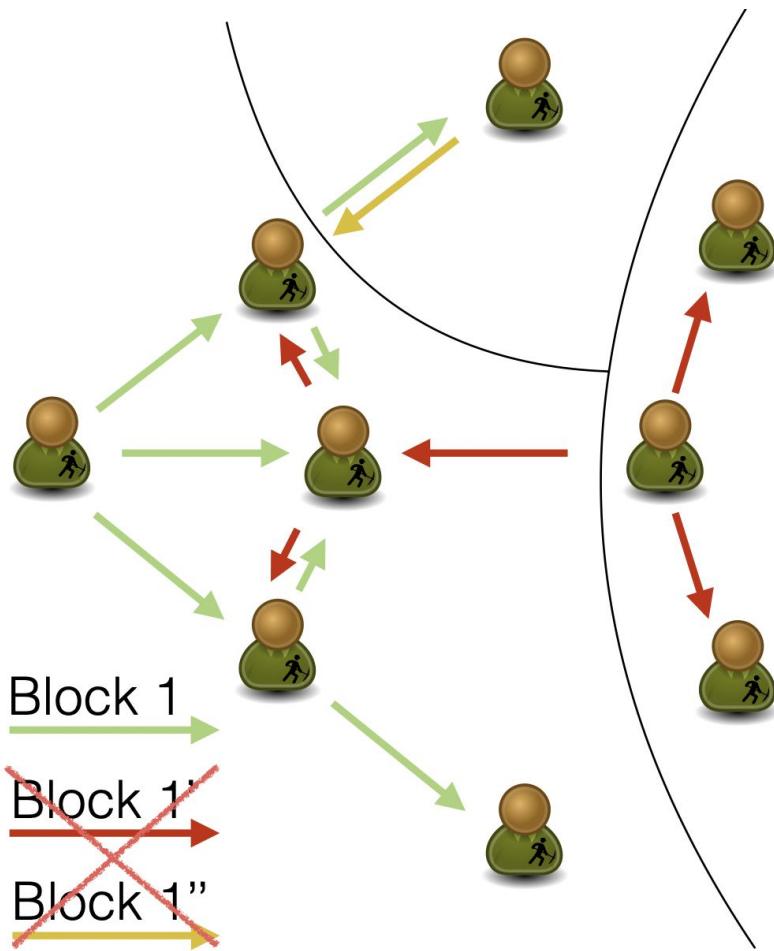
Building blocks

- Miners



Building blocks

- Mining and forks



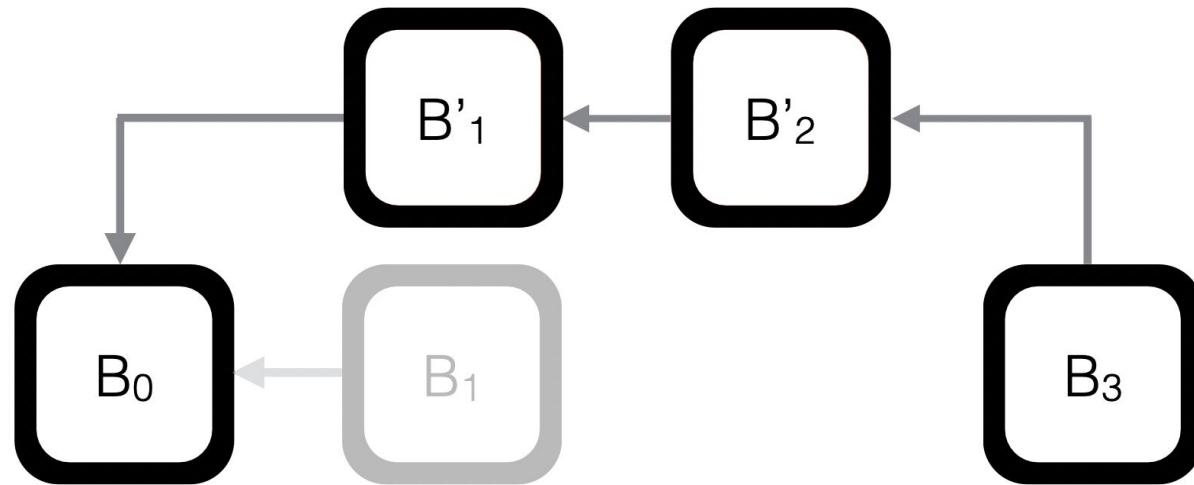
- Network partition
- **Stale blocks = lost efforts**



Selfish Mining
Denial of Service
Double Spending

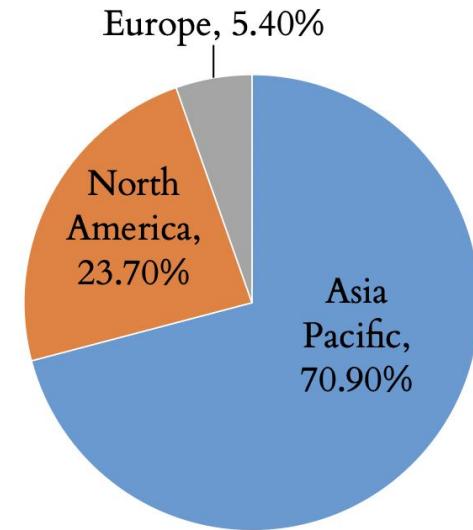
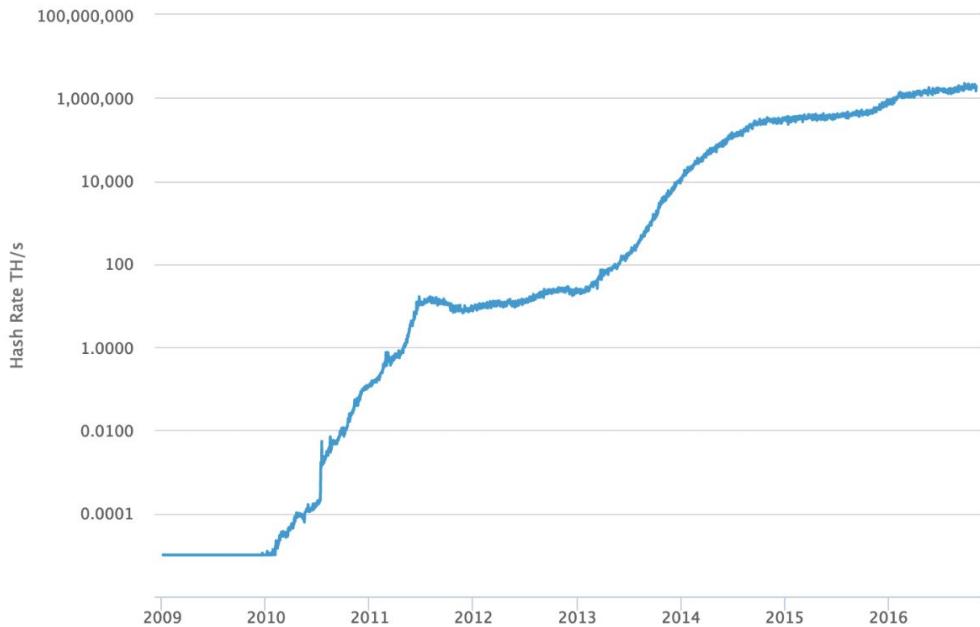
Building blocks

- Blockchain forks



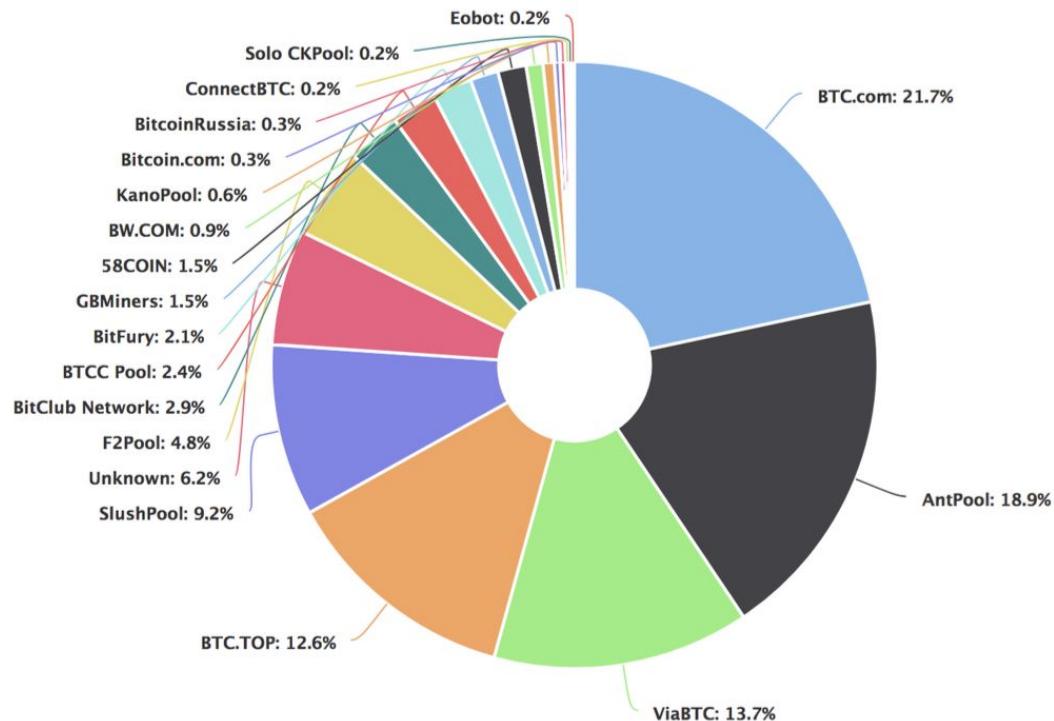
Building blocks

- Mining
 - From CPU to GPU to ASICs



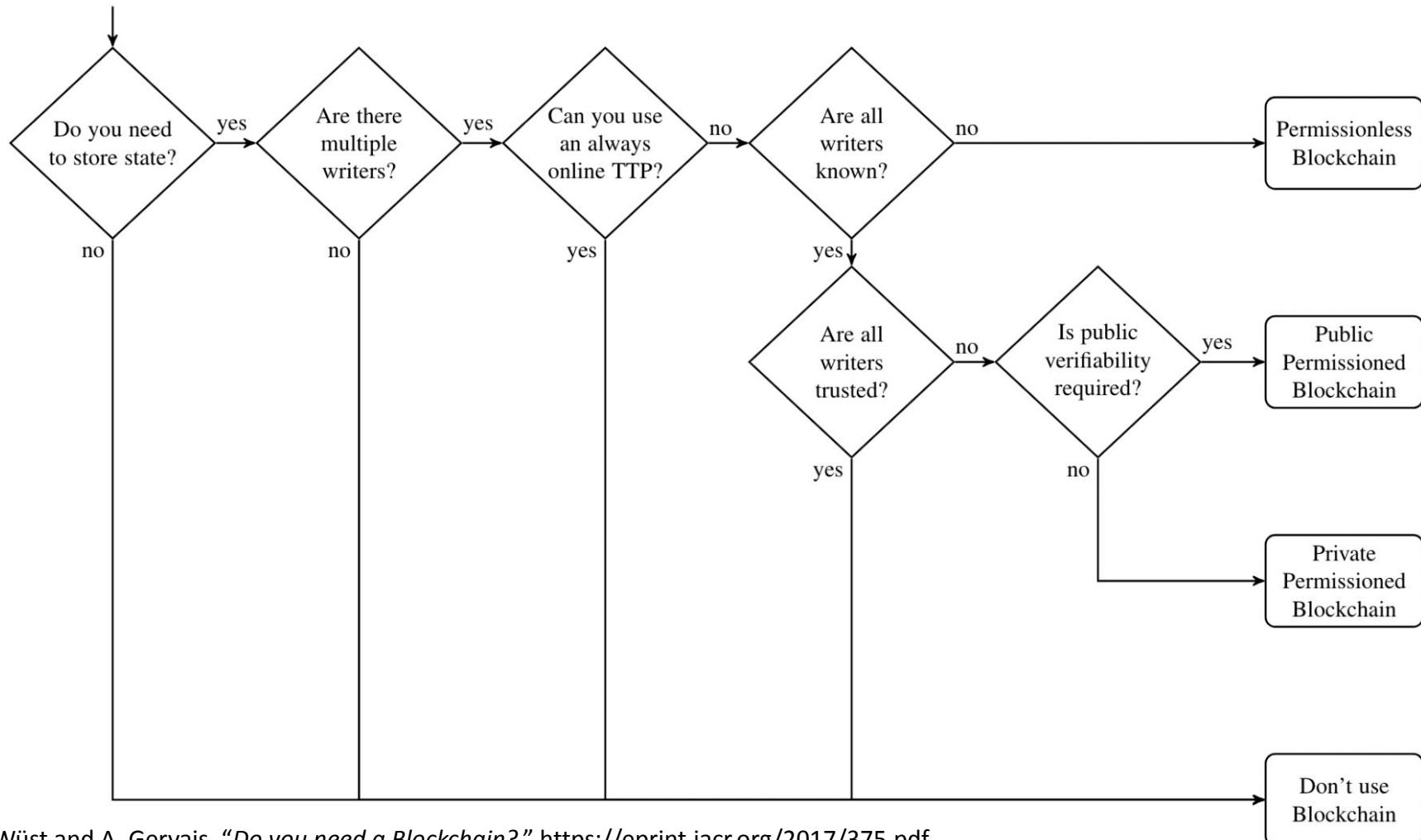
Building blocks

- Mining pools
 - Probability of finding a block alone is very small
 - Unite in Mining pools
 - Payout is done proportional to the work



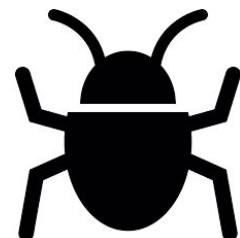
Need of Blockchains

- Possible decision tree



Need of Blockchains

- Typical scenarios
 - International banking
 - Supply chain management
 - Ransomware



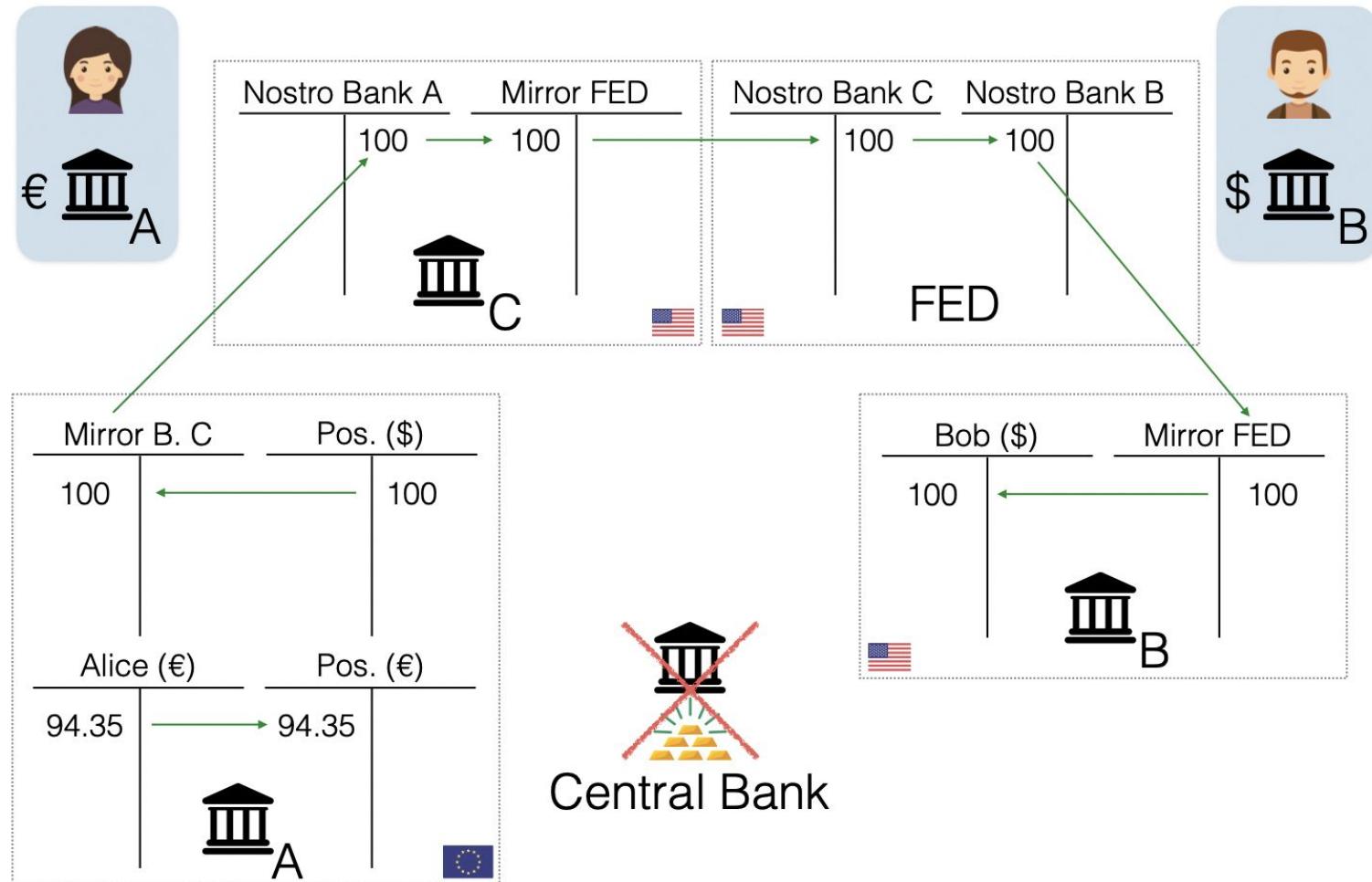
Need of Blockchains

- Bank payments
 - Transfer within a Bank ($A \rightarrow B$)
 - Simple change of books
 - Bank's total balance doesn't change
 - Transfer across Banks



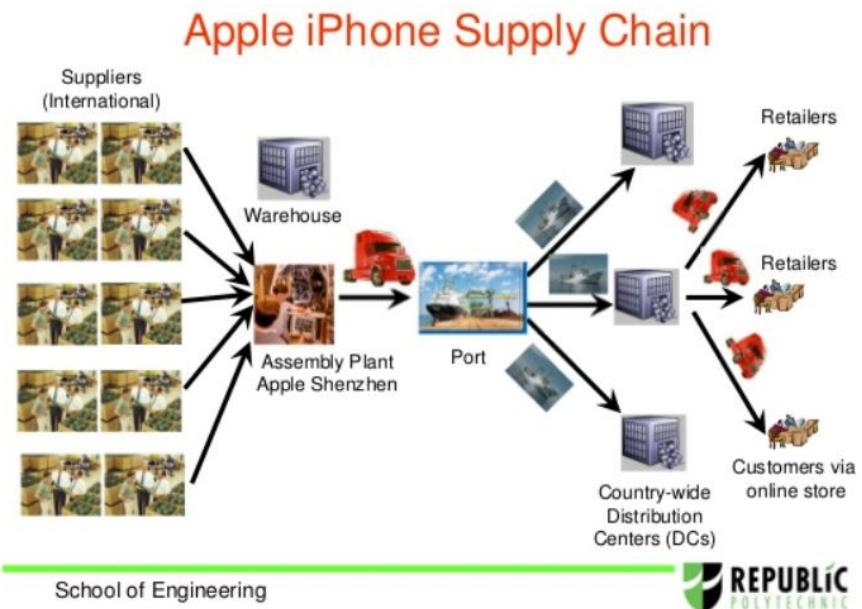
Need of Blockchains

- International payments



Need of Blockchains

- Supply chain management
 - Producing an iPhone involves many parties (> 500 suppliers)



- Supply chain
 - Estimate future demand based on past and current demand
 - What are the fundamental challenges of SCM & blockchain?

Need of Blockchains

- Ransomware

The screenshot shows a web browser displaying an article from The Verge. The URL in the address bar is <https://www.theverge.com/2017/5/12/15630354/nhs-hospitals-ransomware-hack-wannacry>. The page header includes the The Verge logo and navigation links for TECH, SCIENCE, CULTURE, CARS, REVIEWS, LONGFORM, VIDEO, and MORE. Below the header, there are categories for TECH and CYBERSECURITY. The main title of the article is "UK hospitals hit with massive ransomware attack". A subtitle reads "Sixteen hospitals shut down as a result of the attack". The author is listed as Russell Brandom | @russellbrandom | May 12, 2017, 11:36am EDT. There are social sharing buttons for Facebook, Twitter, and LinkedIn. To the right of the main content, there is a "NOW TRENDING" sidebar featuring a thumbnail of a green hill under a blue sky and a headline about Microsoft issuing a patch for Windows XP.

UK hospitals hit with massive ransomware attack

Sixteen hospitals shut down as a result of the attack

by Russell Brandom | @russellbrandom | May 12, 2017, 11:36am EDT

SHARE | TWEET | LINKEDIN

NOW TRENDING

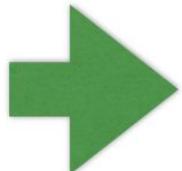
Microsoft issues 'highly unusual' Windows XP patch to prevent massive ransomware attack

Need of Blockchains

- The future of Ransomware
 - Ransomware requires
 - A key to encrypt the victim's data
 - A payment mechanism

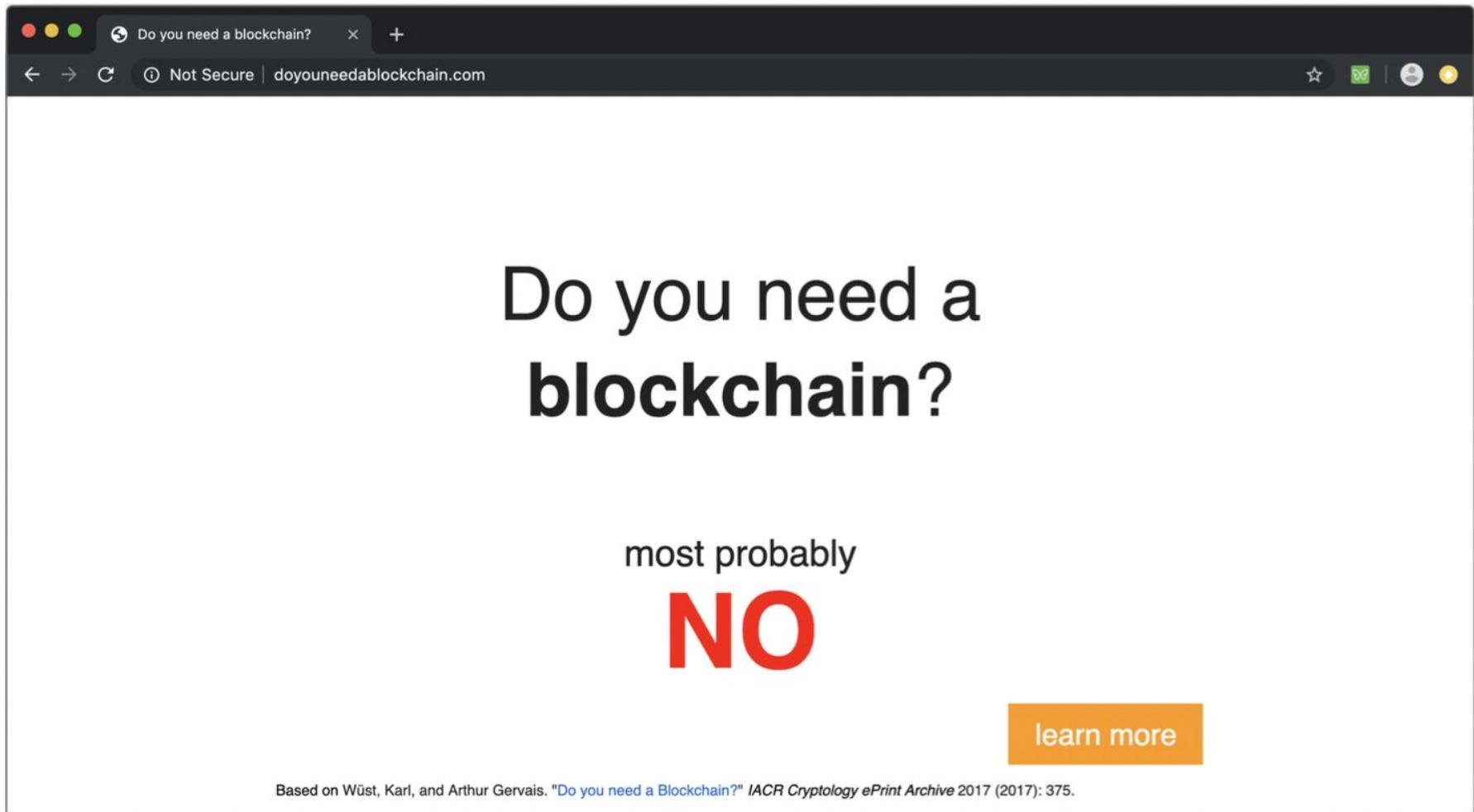
```
if(victim paid) {  
    Reveal key;  
    Victim can decrypt data;  
} else {  
    Don't reveal the key;  
}
```

Done manually!



Autonomous Randomware?

Need of Blockchains



The screenshot shows a web browser window with the following details:

- Title Bar:** Do you need a blockchain?
- Address Bar:** Not Secure | doyouneedablockchain.com
- Content Area:**
 - Main Text:** Do you need a **blockchain?**
 - Text Below:** most probably
 - Large Red Text:** NO
 - Call-to-Action:** learn more (in an orange button)

At the bottom of the page, there is a small note: "Based on Wüst, Karl, and Arthur Gervais. "Do you need a Blockchain?" IACR Cryptology ePrint Archive 2017 (2017): 375."

Scaling Blockchains

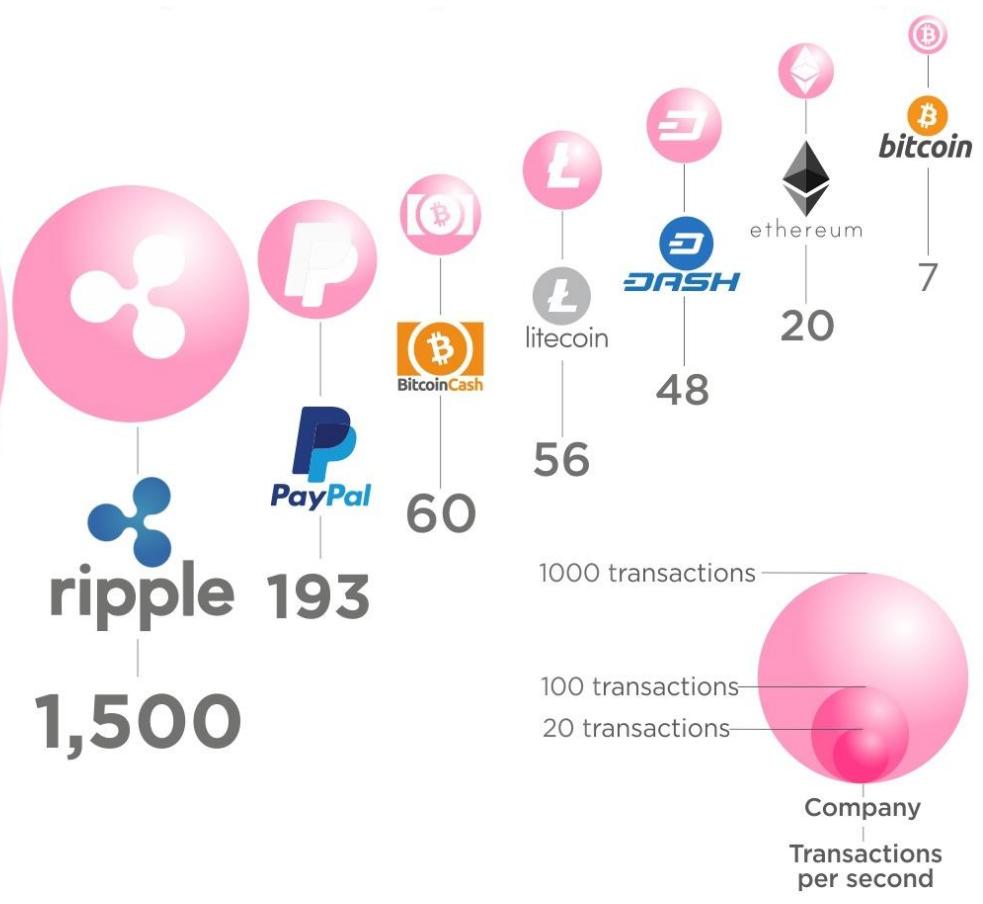
- Cryptocurrencies' transaction speeds compared to VISA & Paypal



24,000

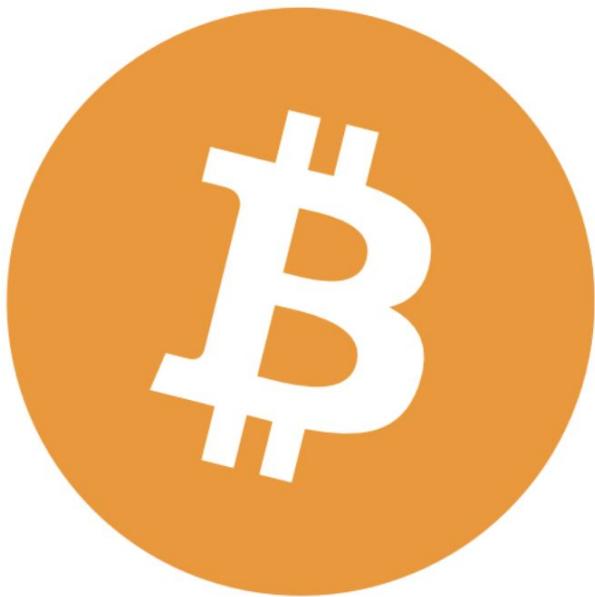
Article & Sources:

<https://howmuch.net/articles/crypto-transaction-speeds-compared>
<https://howmuch.net/sources/crypto-transaction-speeds-compared>



Scaling Blockchains

- Cryptocurrencies do not scale



Size of tx in KB

~ 10 tps



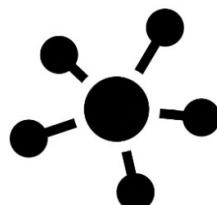
Complexity of tx in “Gas”

Scaling Blockchains

- Why doesn't it scale?



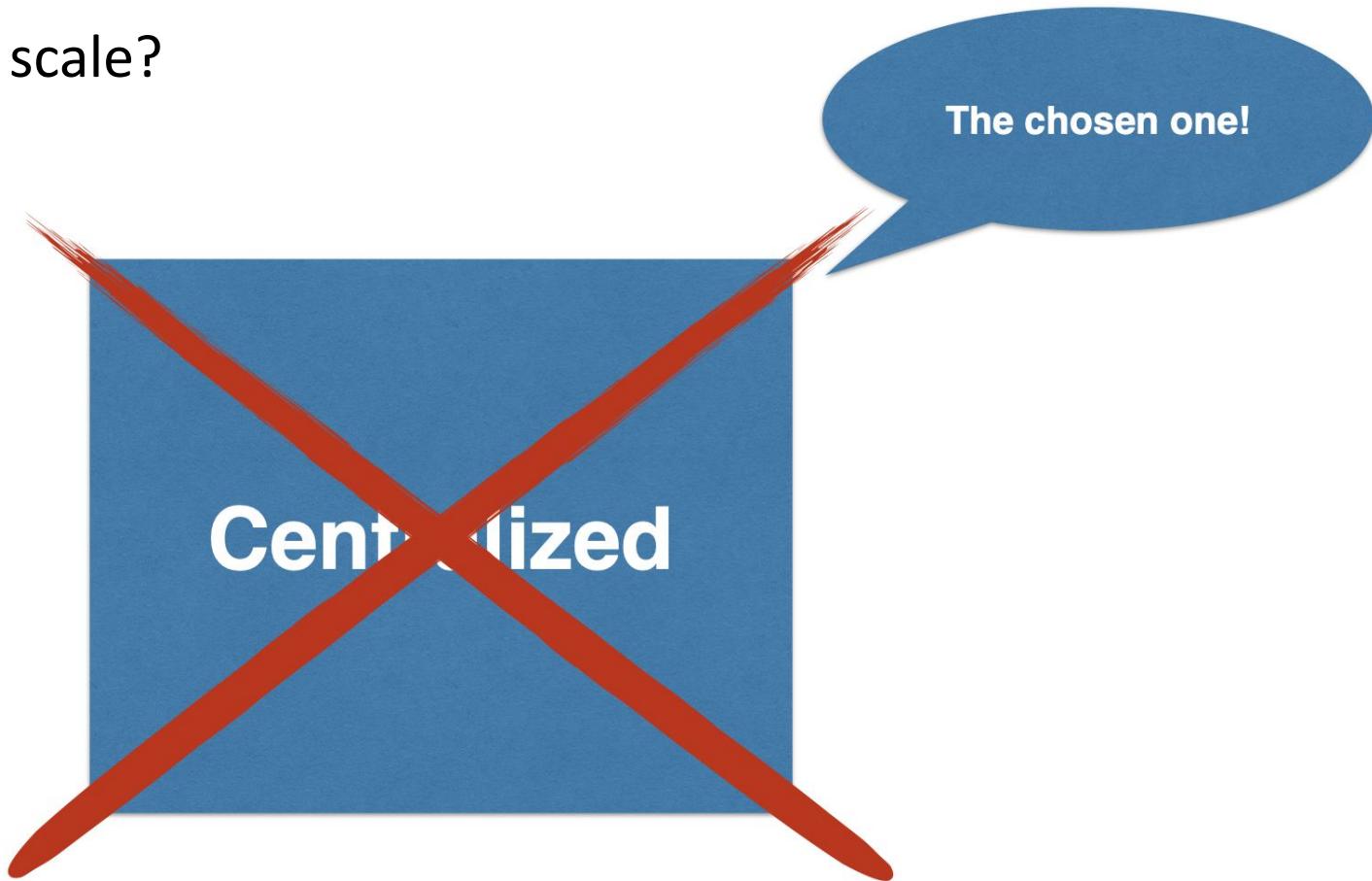
Many users



Many validators

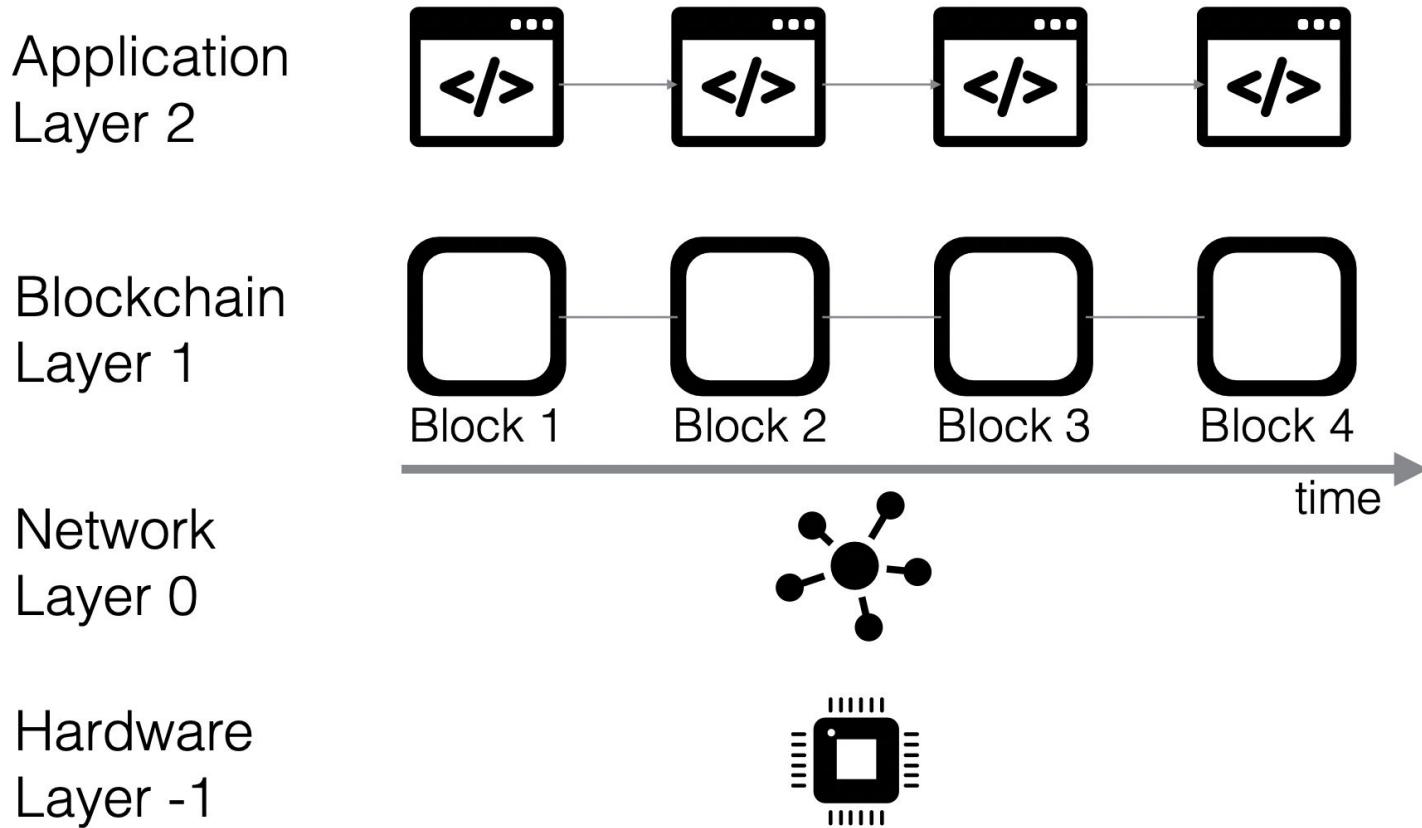
Scaling Blockchains

- How can we scale?



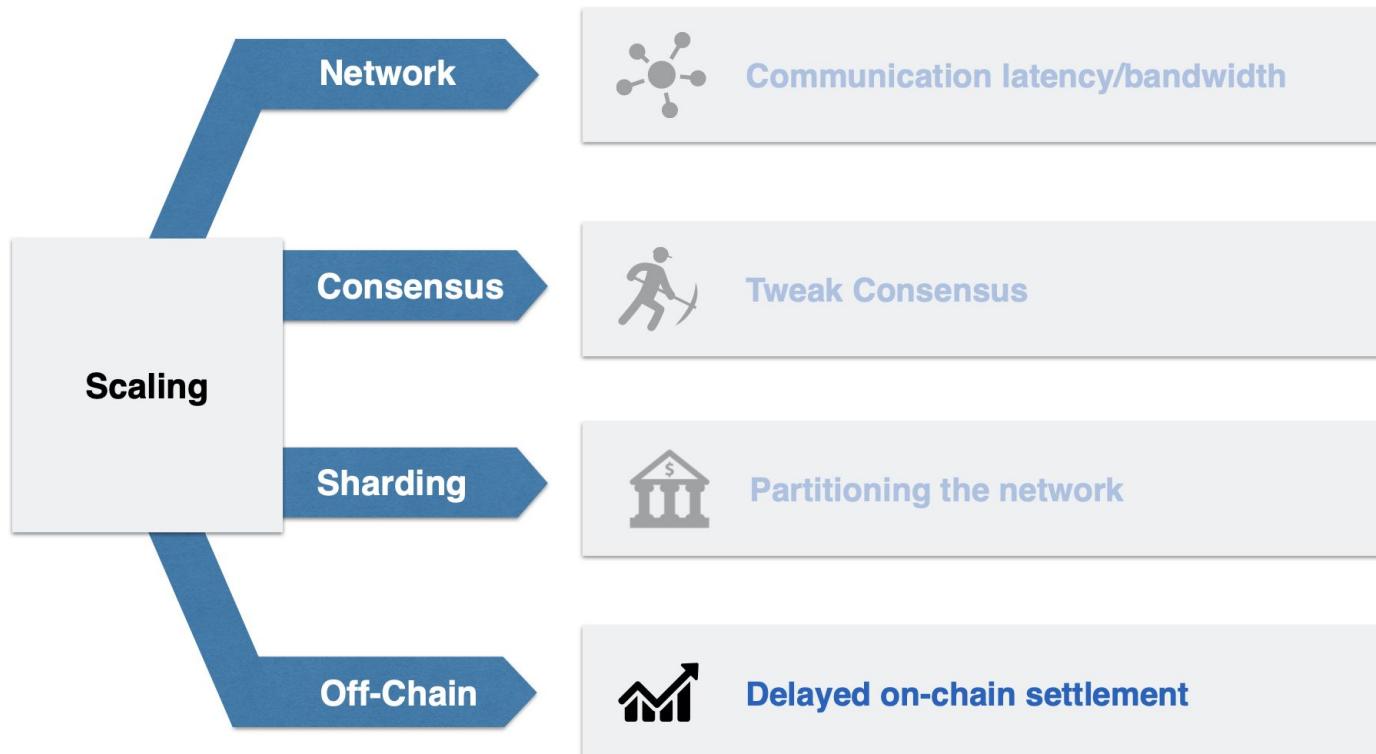
Scaling Blockchains

- Blockchain layers



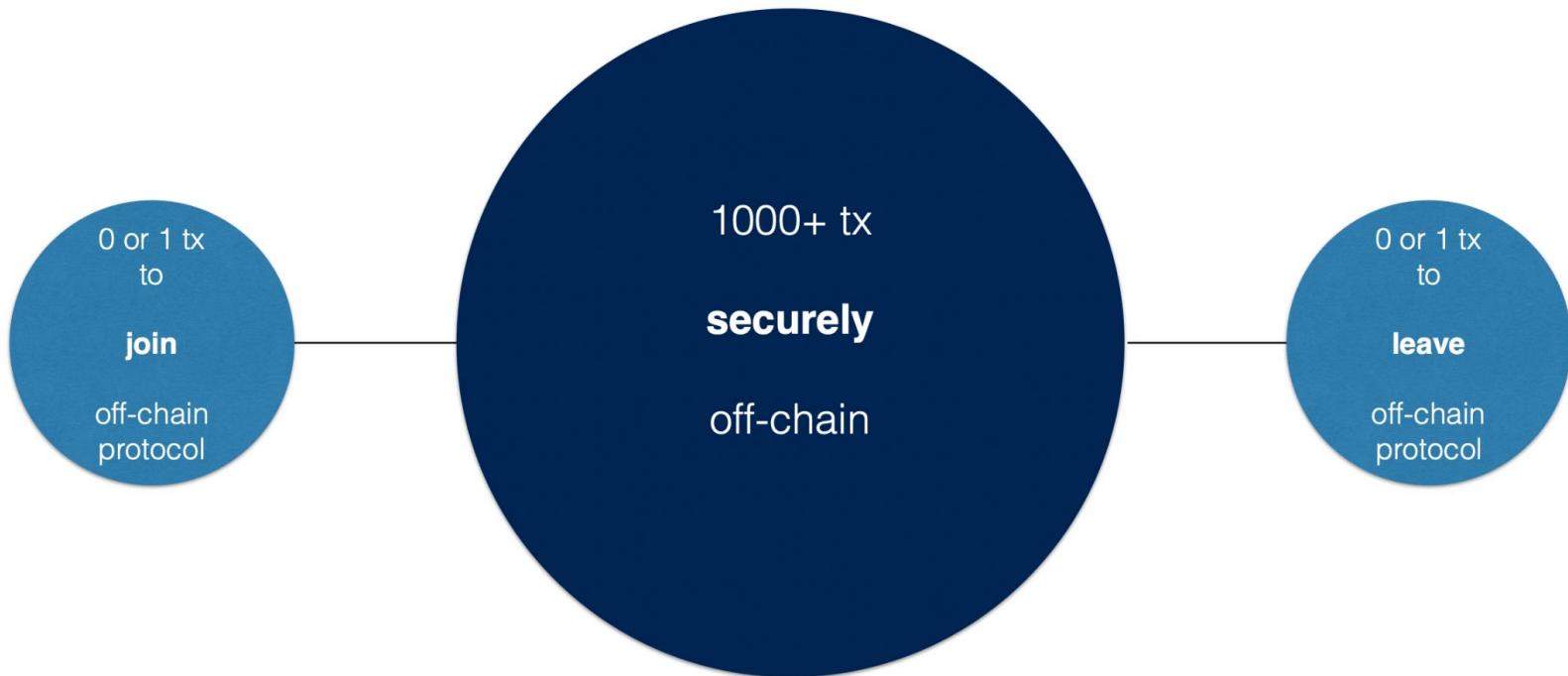
Scaling Blockchains

- How can we scale?



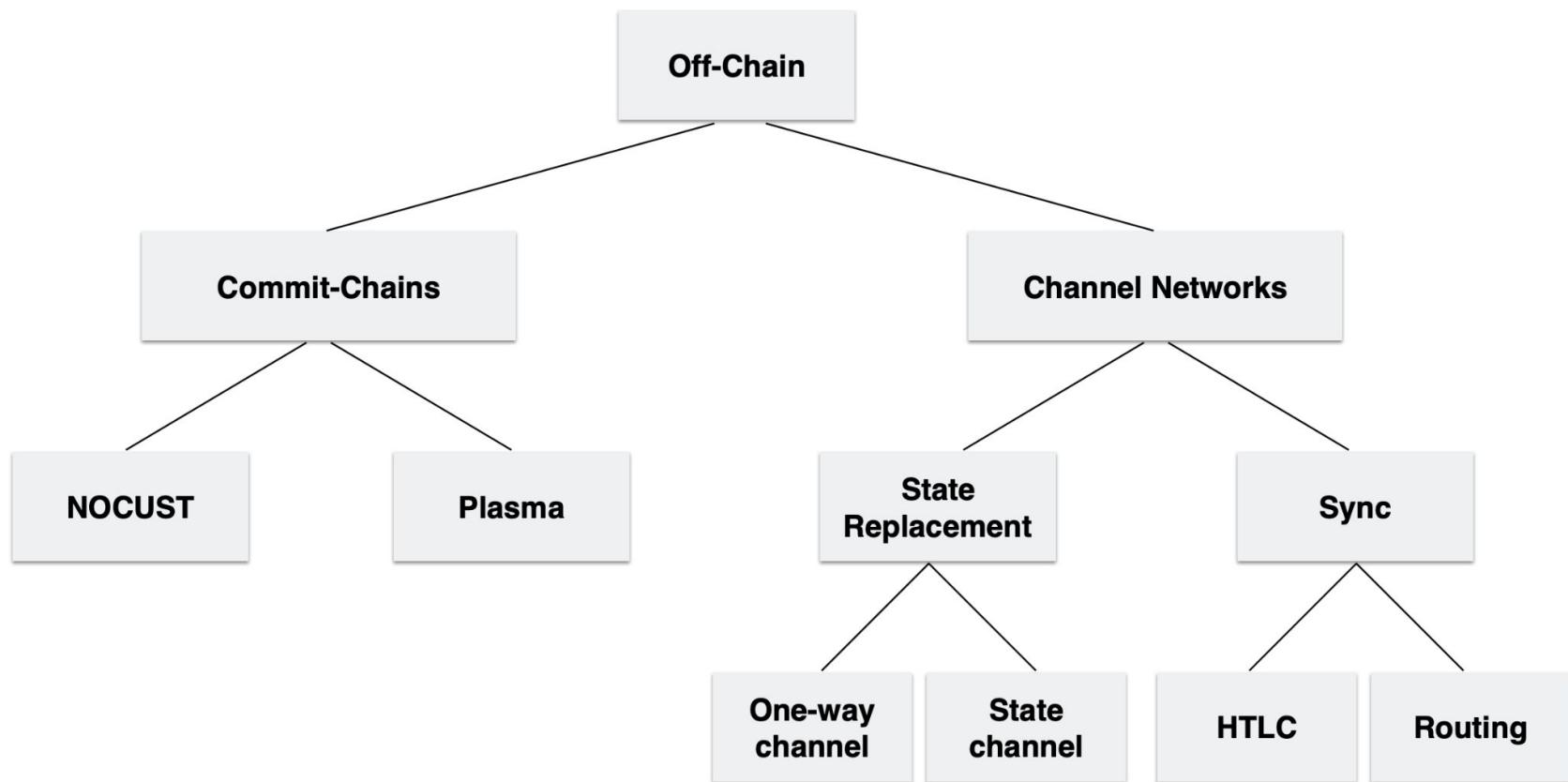
Scaling Blockchains

- Why is off-chain exciting?
 - No consensus latency or mining fees
 - While still achieving non-custodial security
 - Backward compatibility!



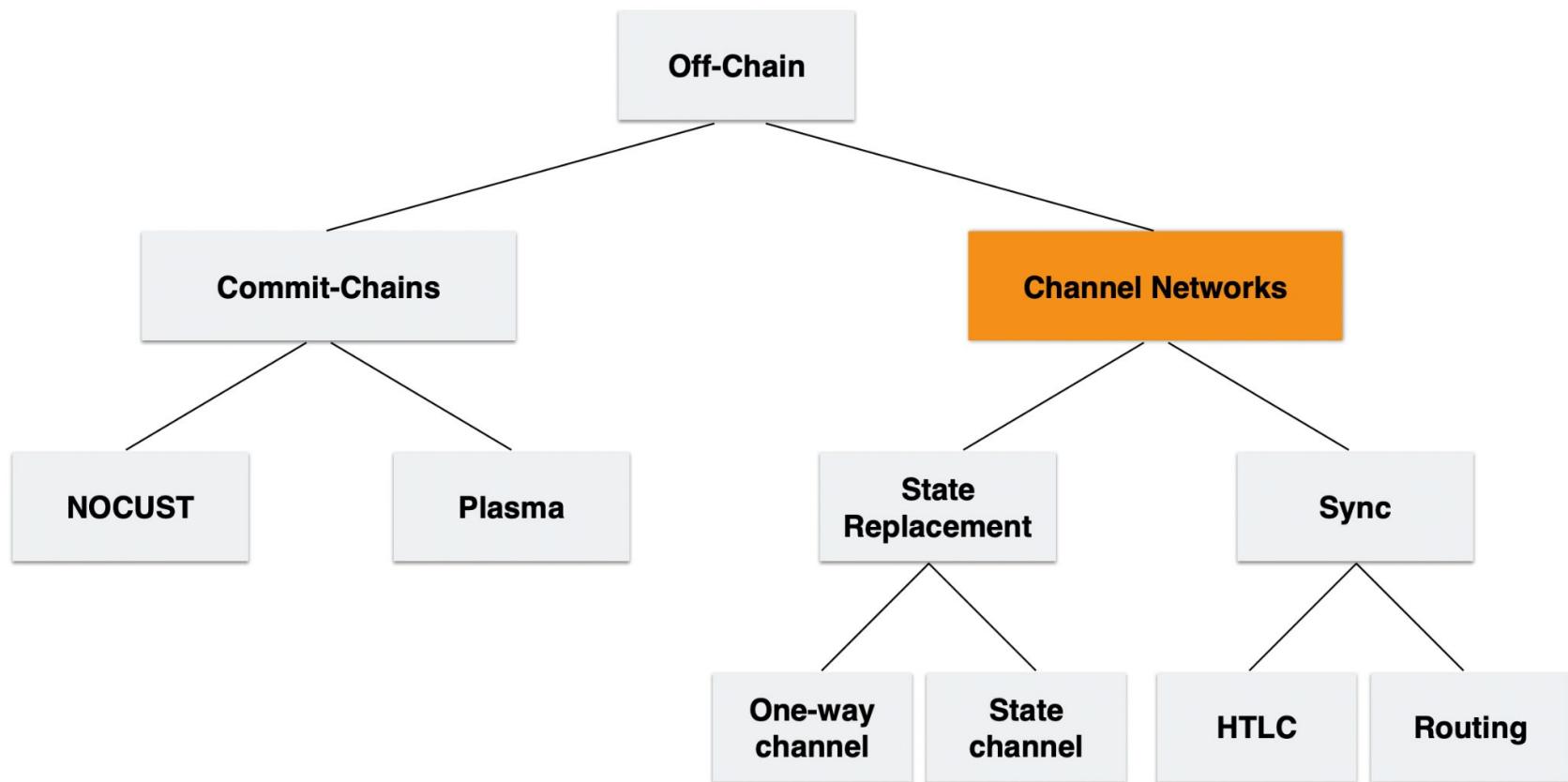
Scaling Blockchains

- Which off-chain solution?



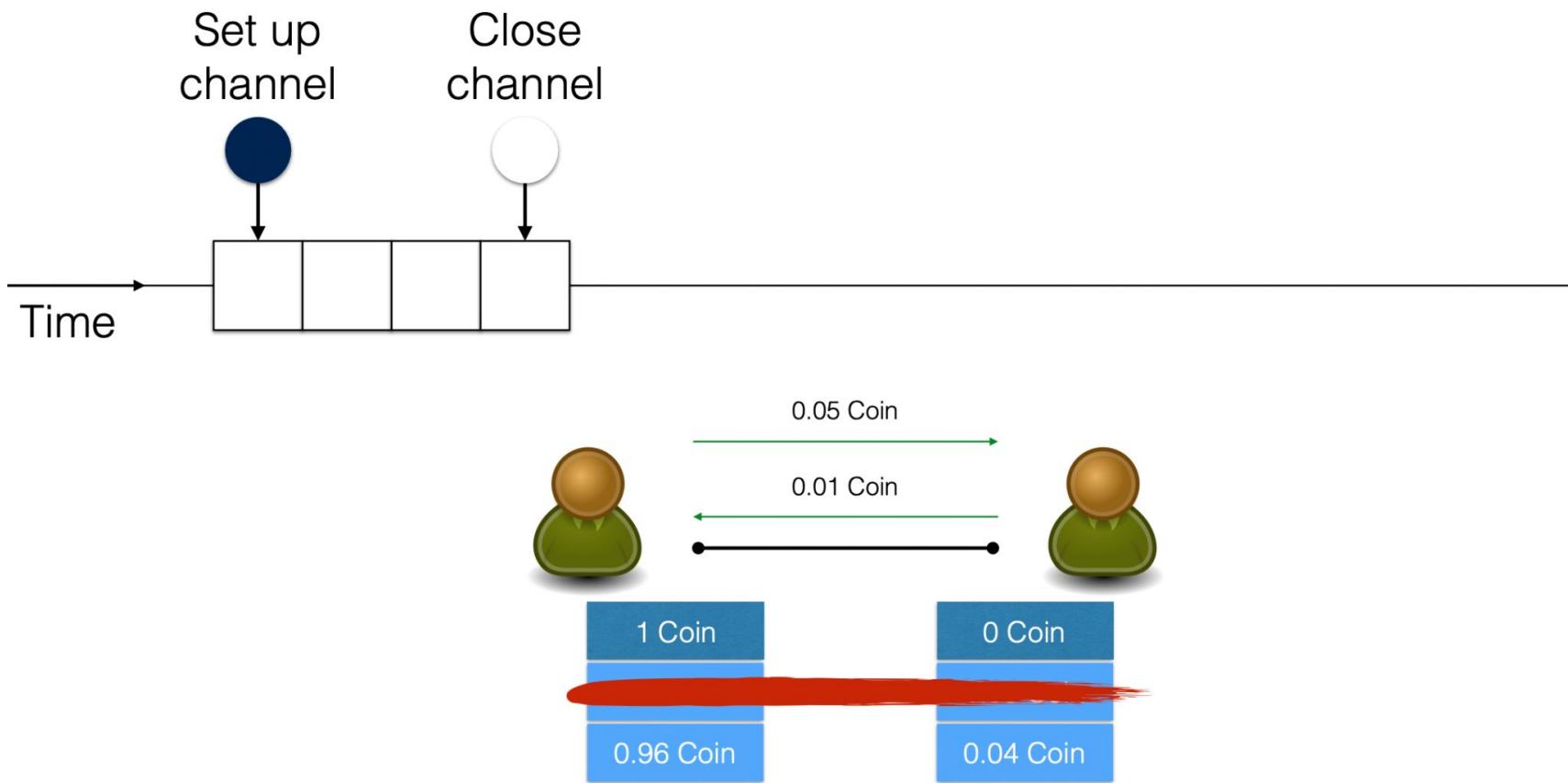
Scaling Blockchains

- Which off-chain solution?



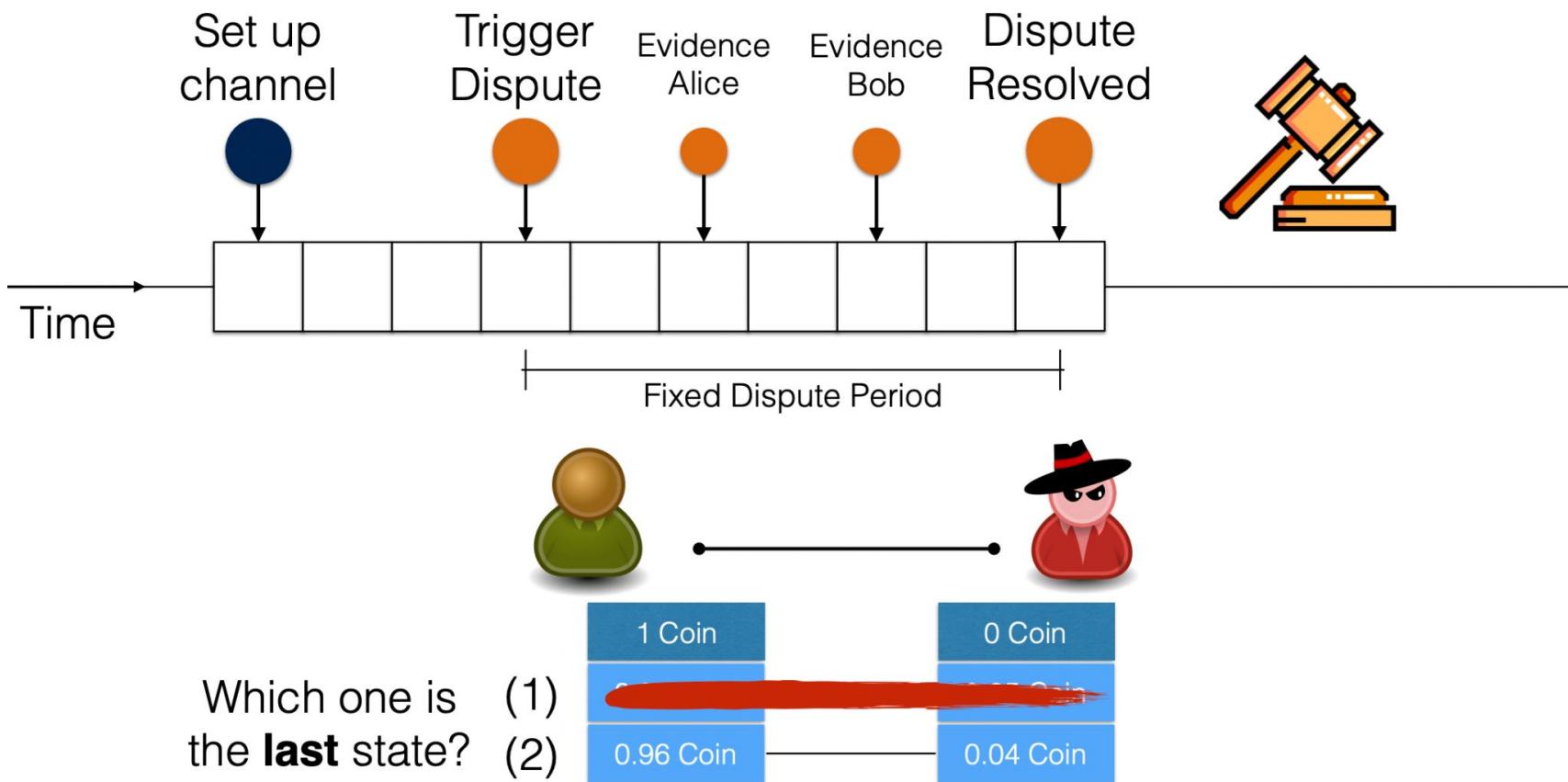
Scaling Blockchains

- Payment channel



Scaling Blockchains

- Payment channel
 - Dispute



Scaling Blockchains

- Payment channel

Spilman

Duplex
Micropayment

Lightning

- State channel (game, voting, auctions, etc.)

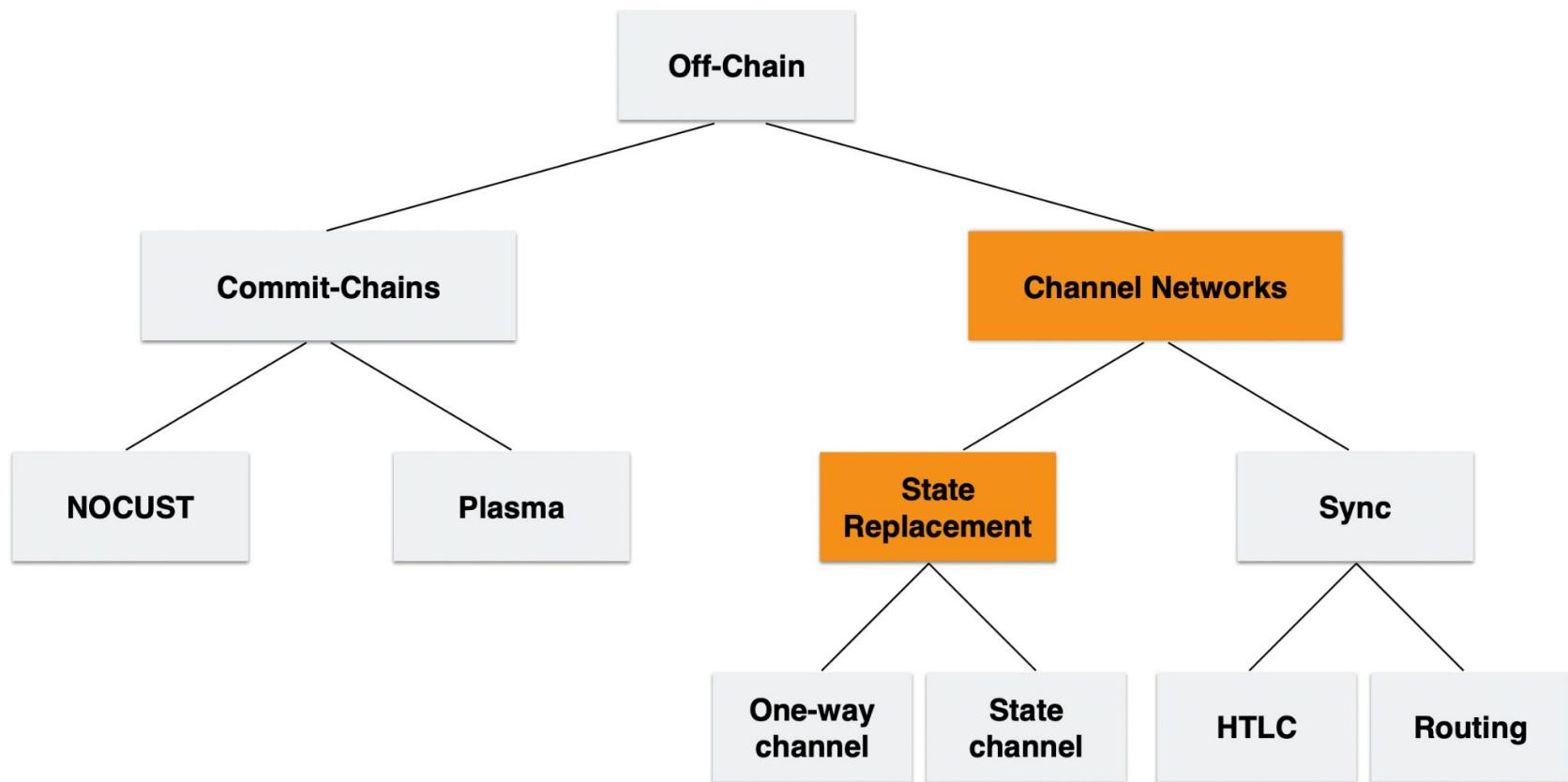
Sprites

Perun

Kitsune

Scaling Blockchains

- Which off-chain solution?



Scaling Blockchains

- State replacement techniques
 - Replace by

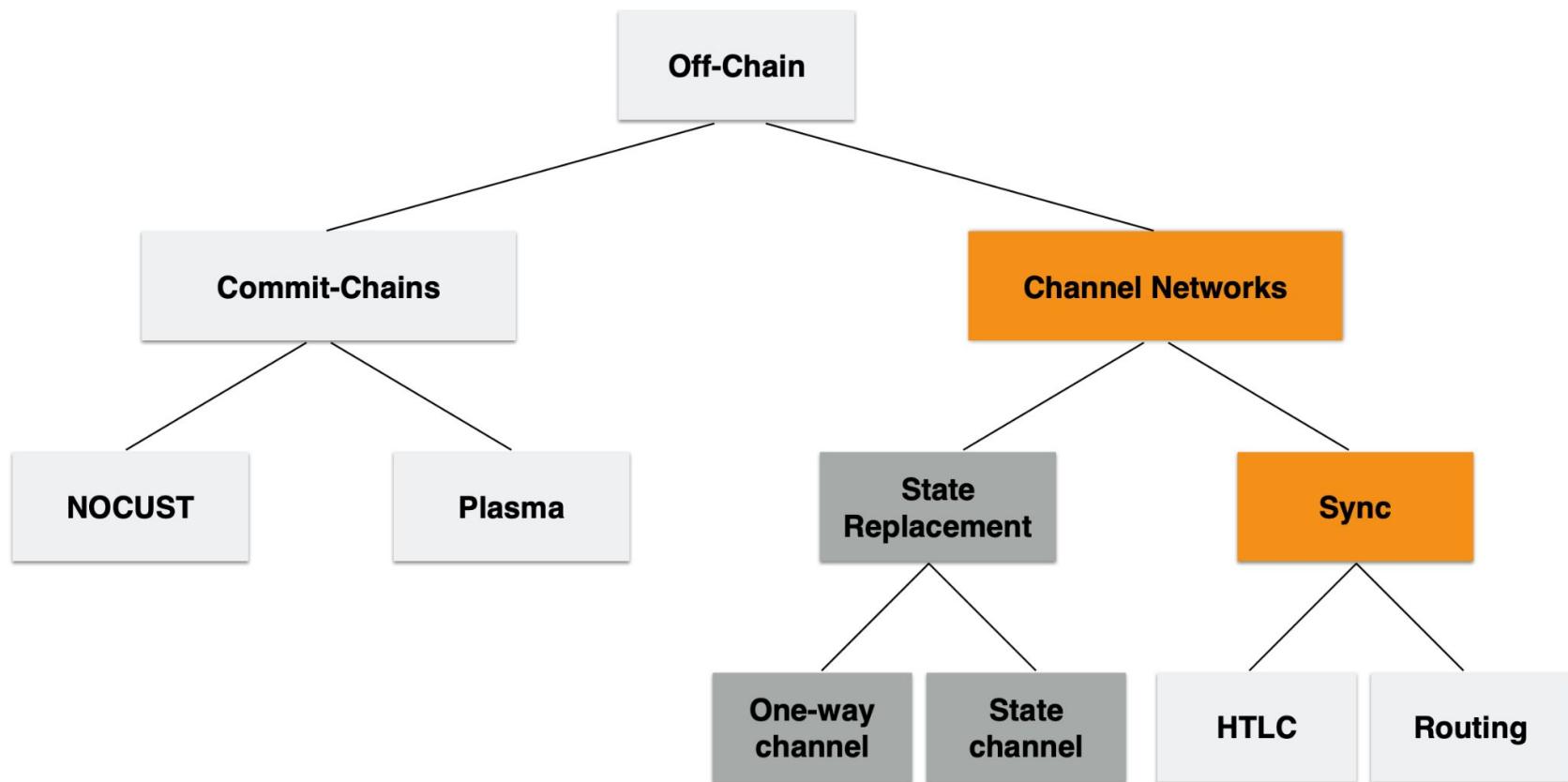
Incentive	→	One-way payments
Time Lock	→	Bi-directional
Revocation	→	Bi-directional, no expiry
Version	→	State change, everyone signs

Protocol
Complexity



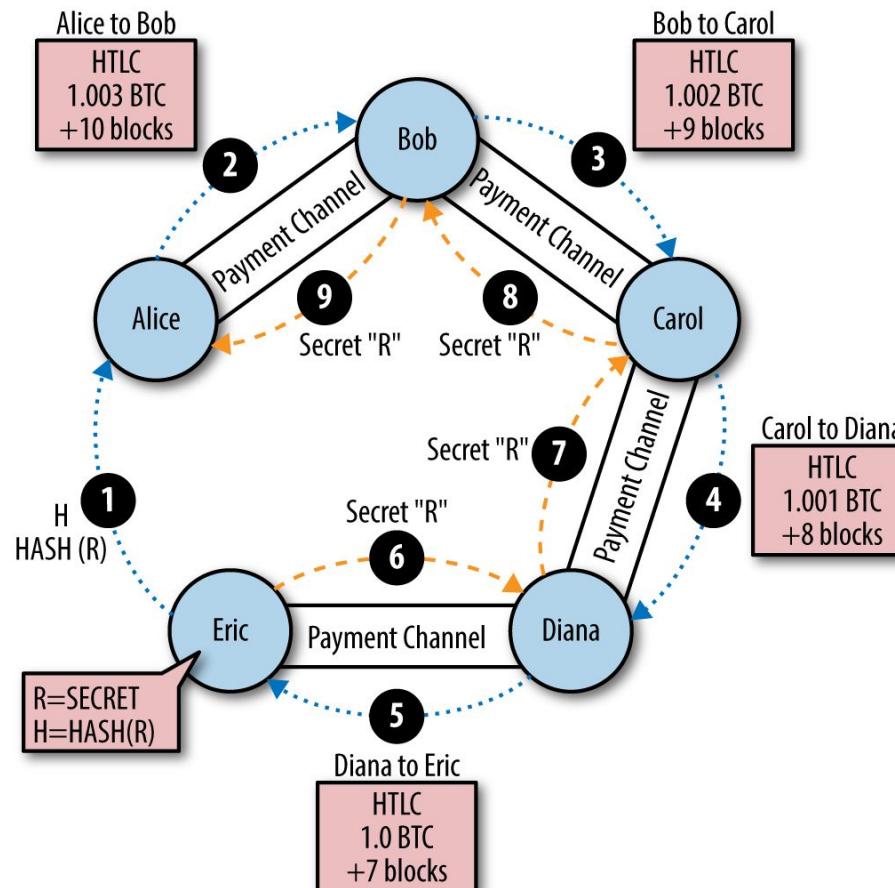
Scaling Blockchains

- Which off-chain solution?



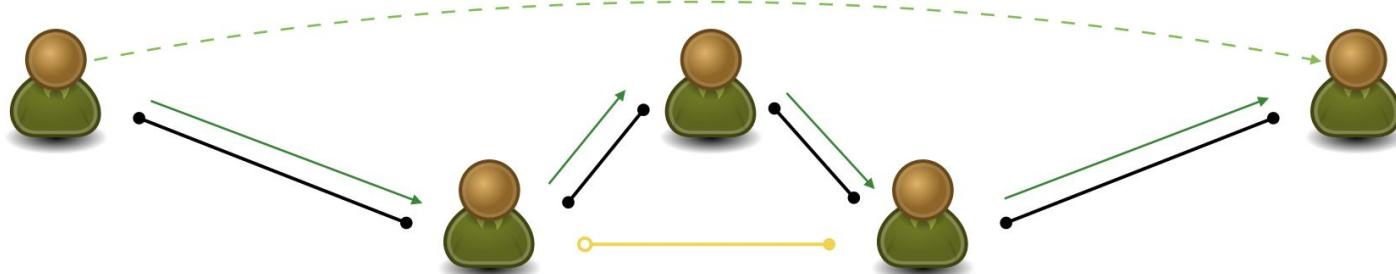
Scaling Blockchains

- Payment channel
 - Hash Time Locked Contracts (HTLC)



Scaling Blockchains

- Payment channel
 - Routing



HTLC + Route Finding

Scaling Blockchains

- The good, bad, and ugly of payment channels



Collateral for each hop



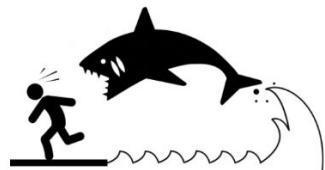
No direct connection needed



Decentralized, limited censorship



On-chain channel establishment



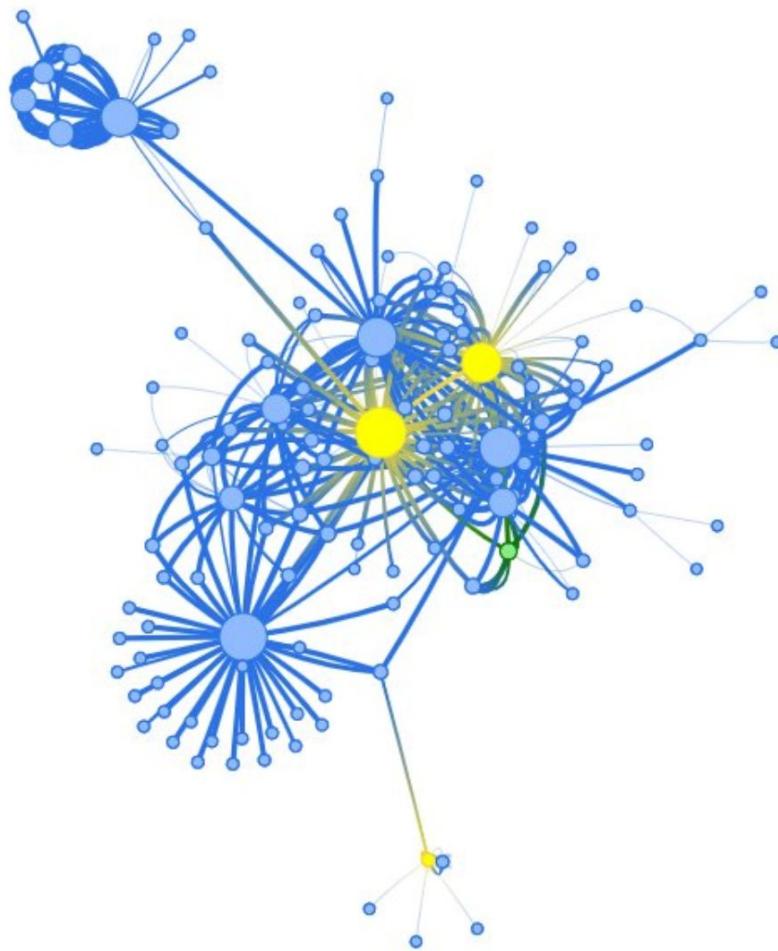
Wormhole attacks



Optimistically fast and cheap

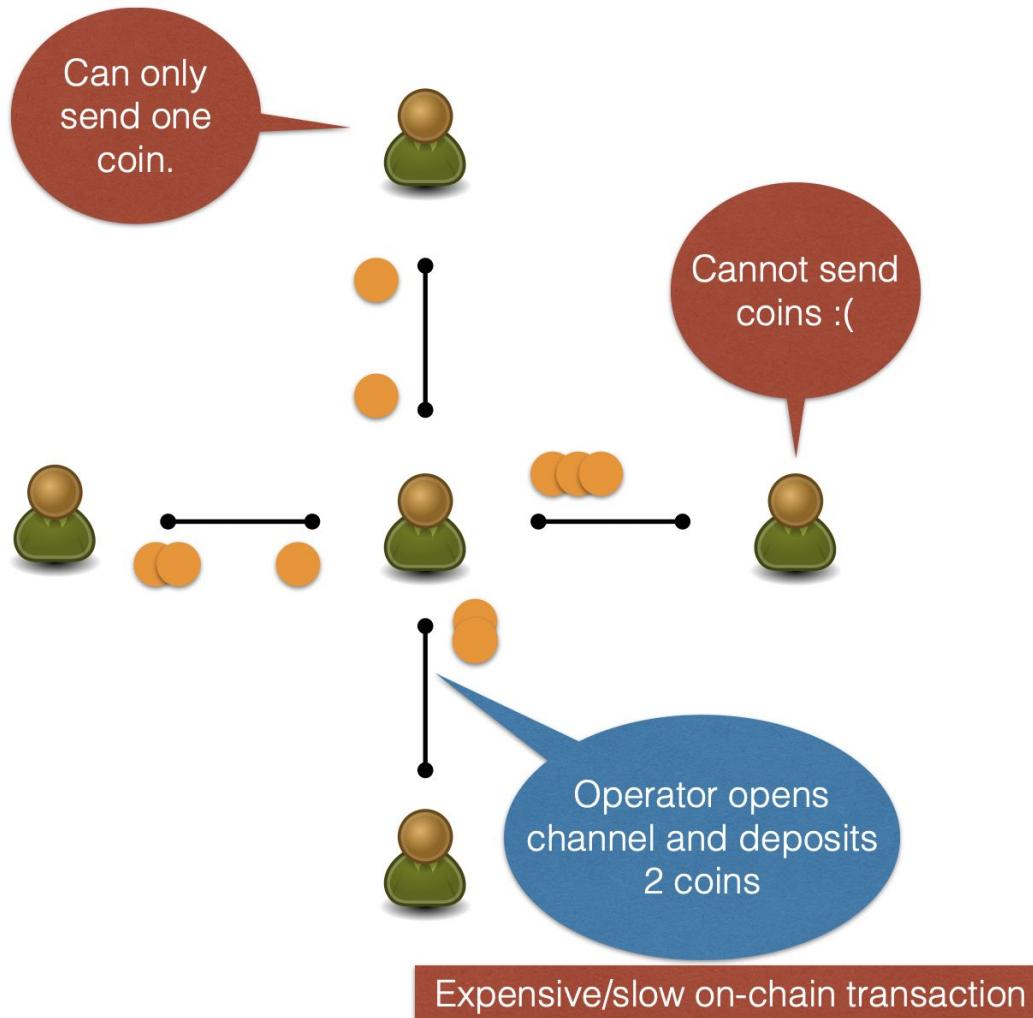
Scaling Blockchains

- Star topologies in theory & practice



Scaling Blockchains

- Payment channel hub

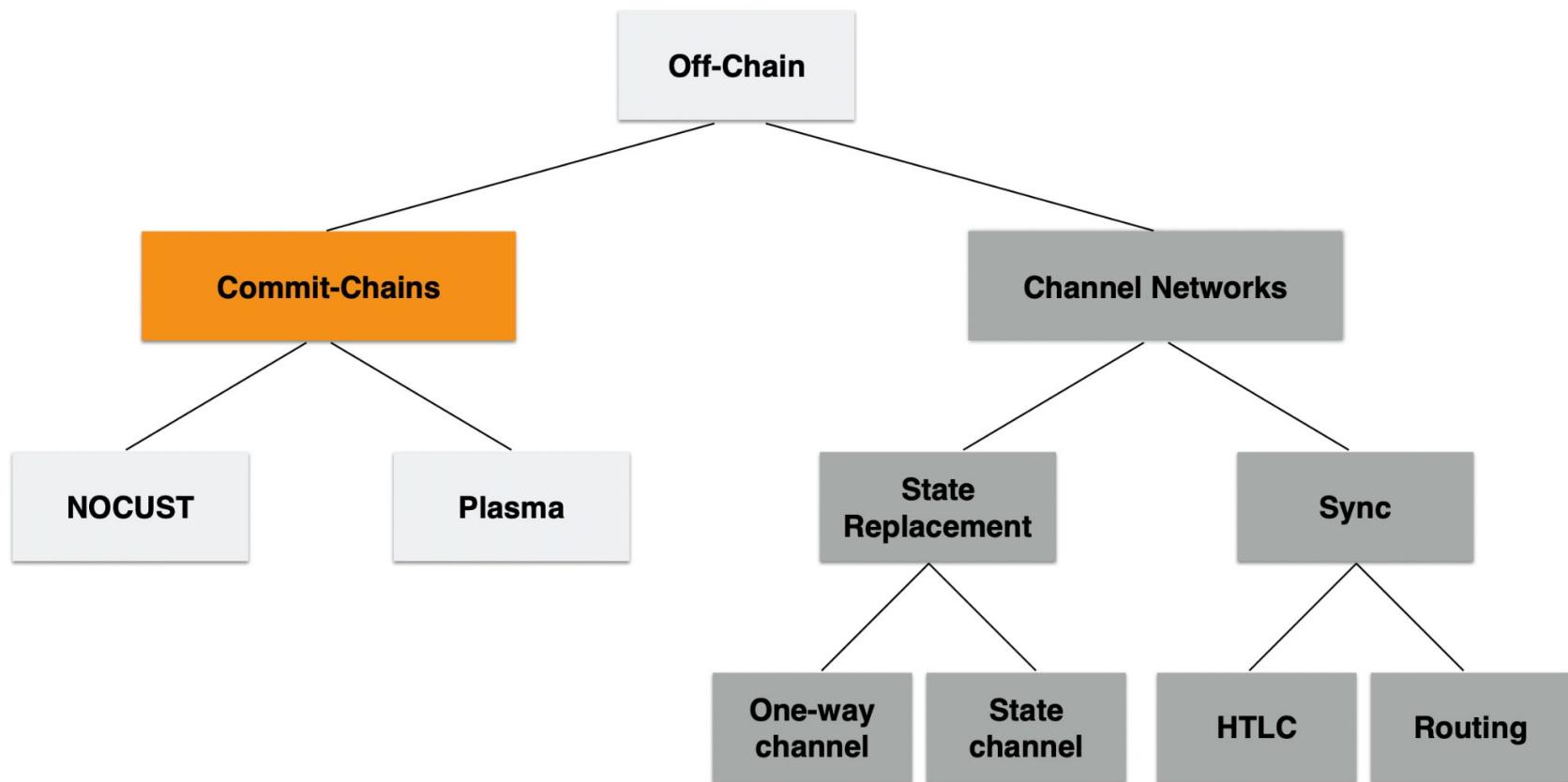


Scaling Blockchains

- Payment channel hub
 - Are great for instant finality and no trust
 - But, expensive to run
 - Can we do better with eventual finality?

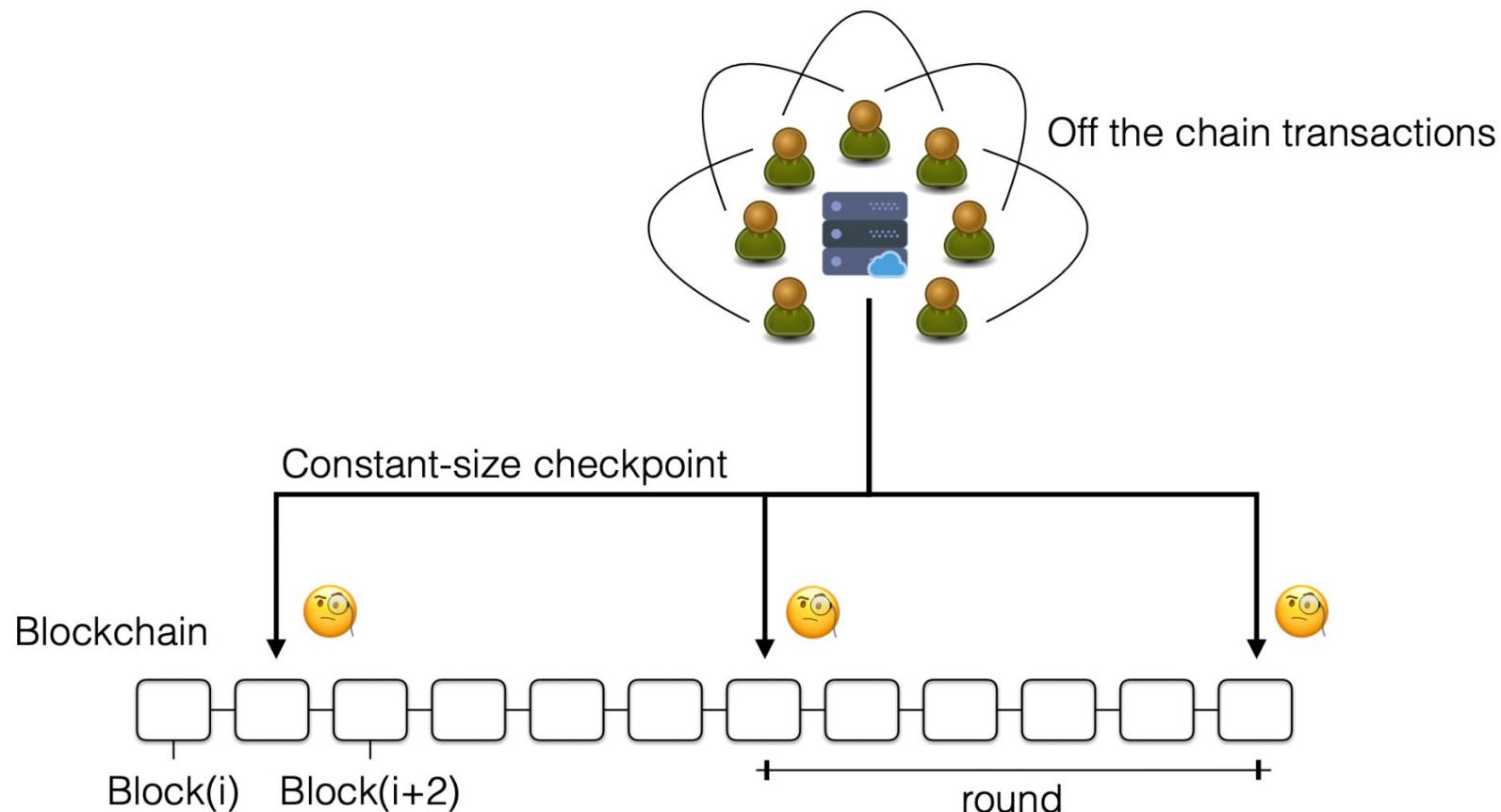
Scaling Blockchains

- Which off-chain solution?



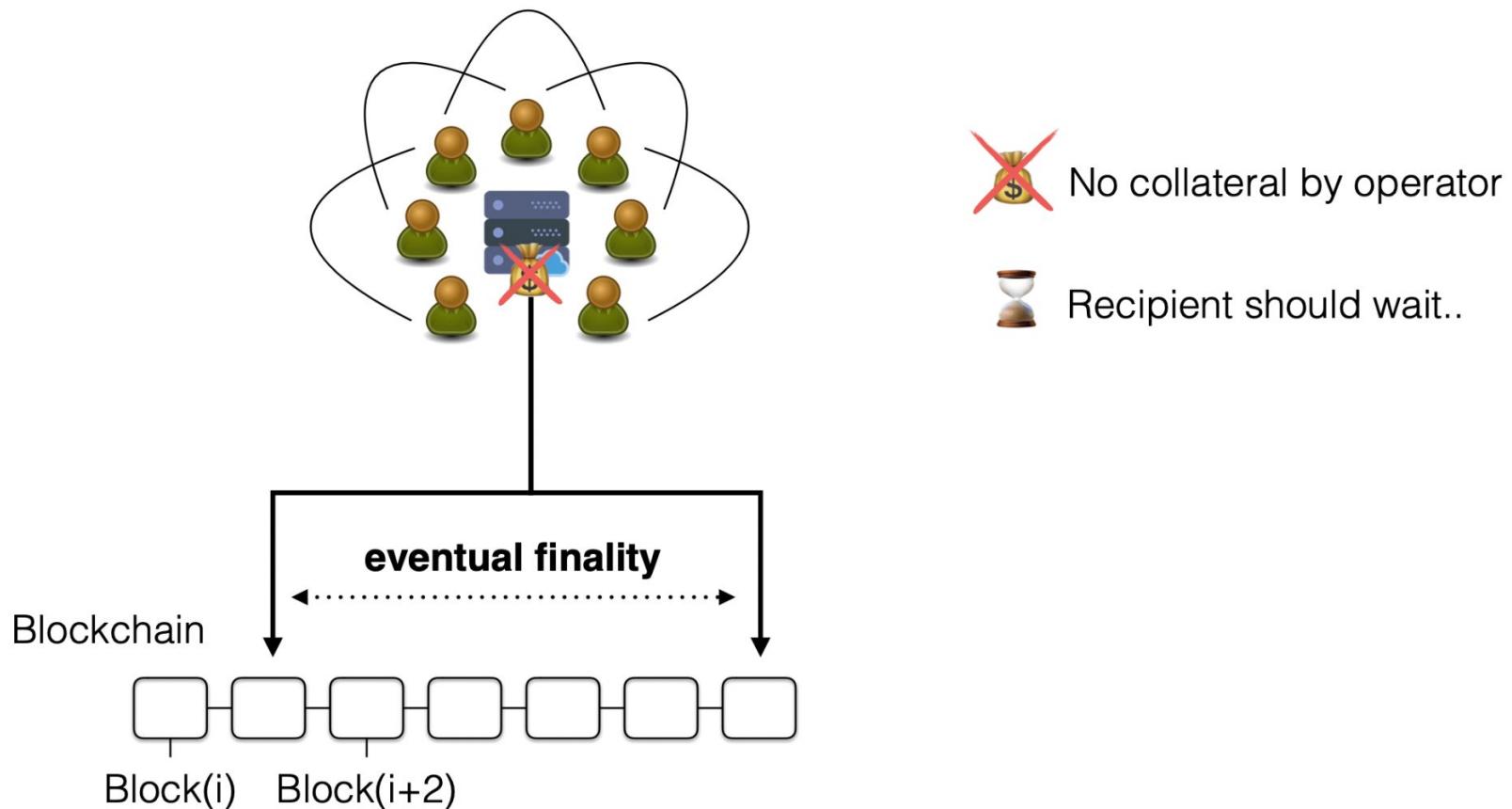
Scaling Blockchains

- Commit-Chains



Scaling Blockchains

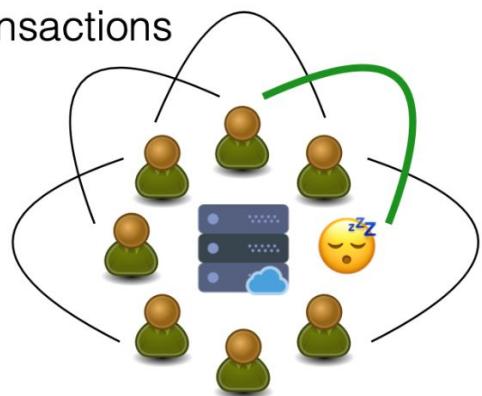
- Without collateral —> Eventual finality



Scaling Blockchains

- Receive TX while offline

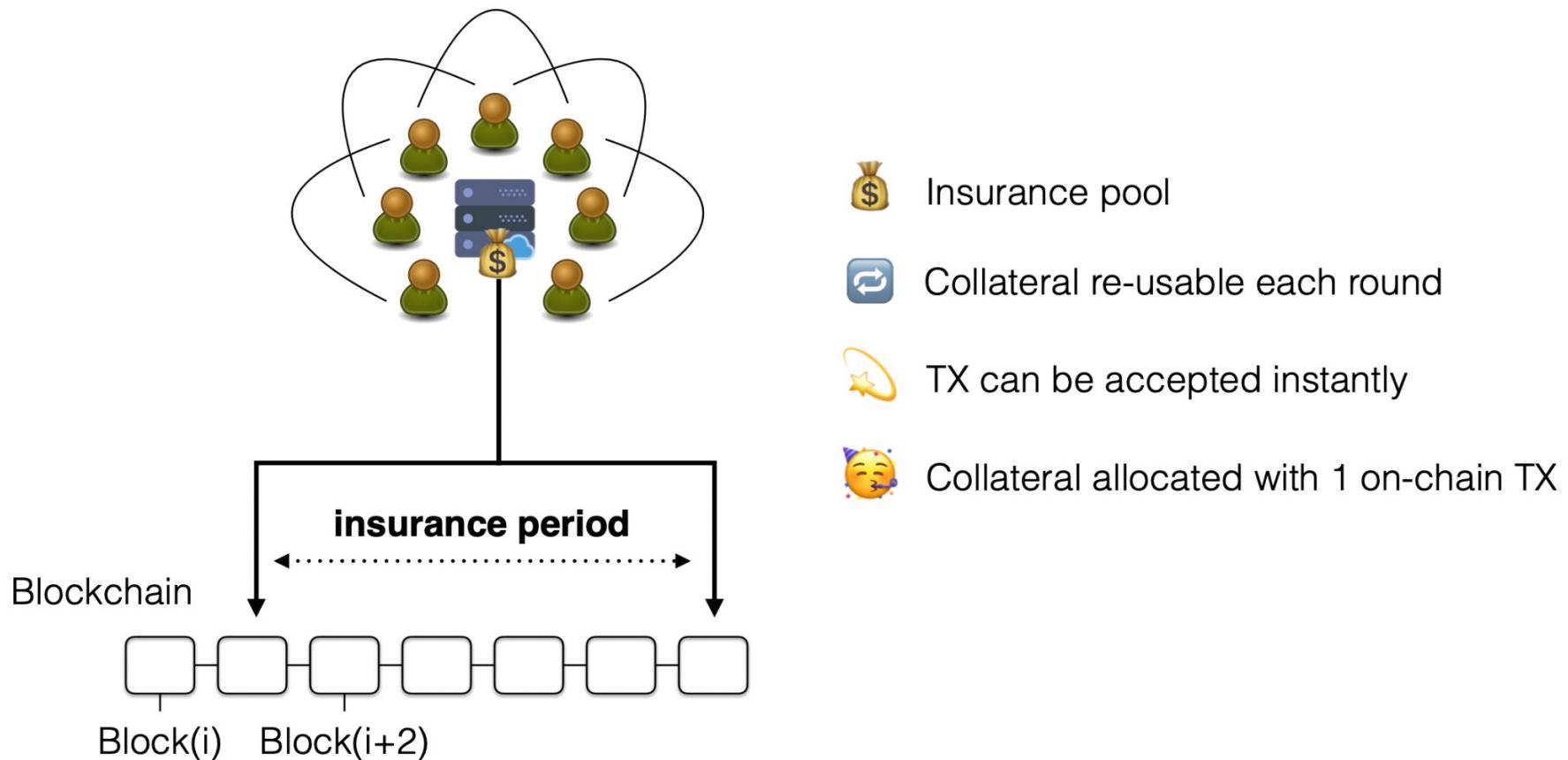
Off the chain transactions



Like on-chain transactions..

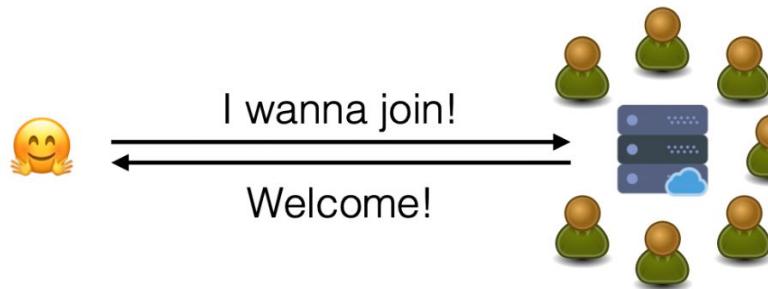
Scaling Blockchains

- With collateral → Instant finality



Scaling Blockchains

- Join without on-chain Transaction



- ◆ Instant
- ◆ CO₂ friendly (Zero gas costs)

Scaling Blockchains

NOCUST

Off-chain ledger state:
user balances



Plasma (Cash)

Off-chain ledger state:
serial number coins

Fungible payments

Slower delayed finality

Lightweight clients

Instant finality support

Atomic Off-chain Swap (TEX)

Non-Fungible payments

Rapid delayed finality

Large amounts of history data

No instant finality support

No known feasibility for swaps

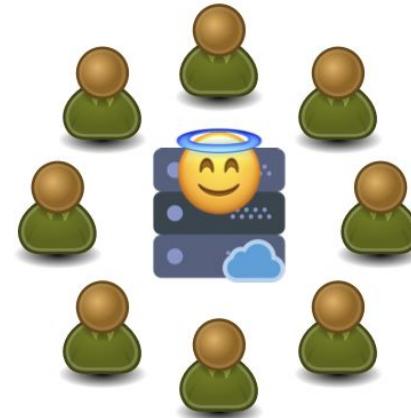
Scaling Blockchains

- A centralized operator
 - Untrusted and non-custodial!



Can

Censor off-chain tx



Cannot

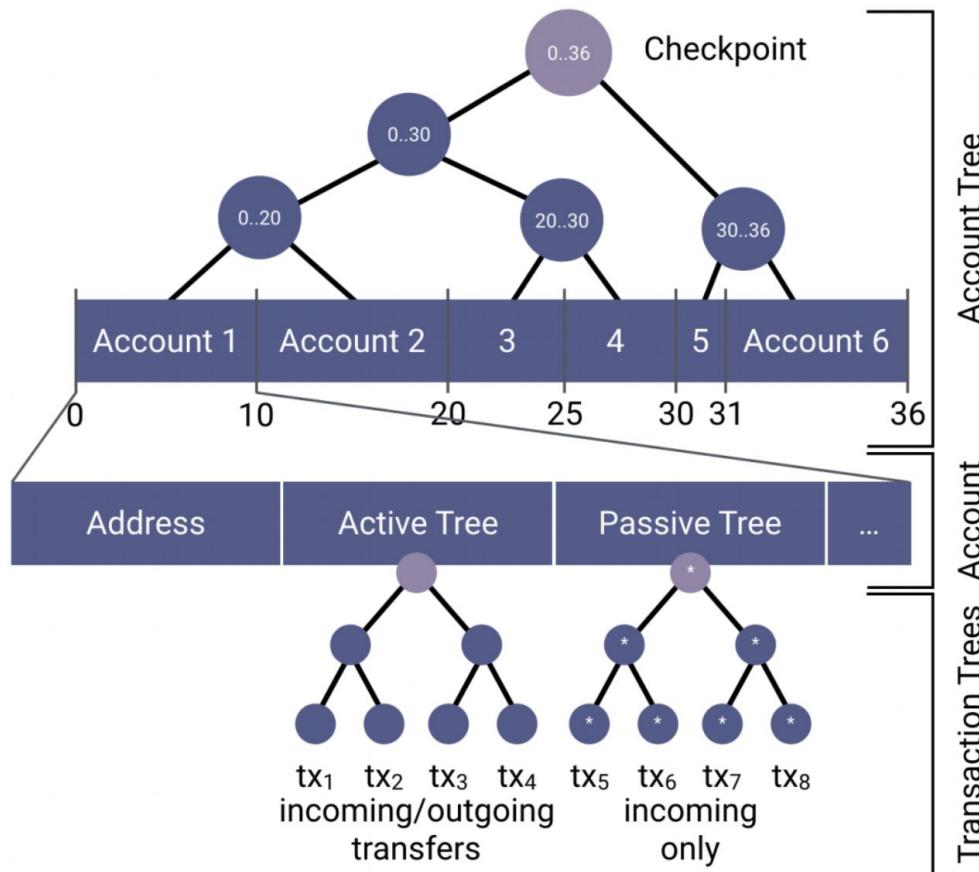
Steal coins

Double spend coins

Mint new coins

Scaling Blockchains

- NOCUST key innovation
 - Merkle interval tree



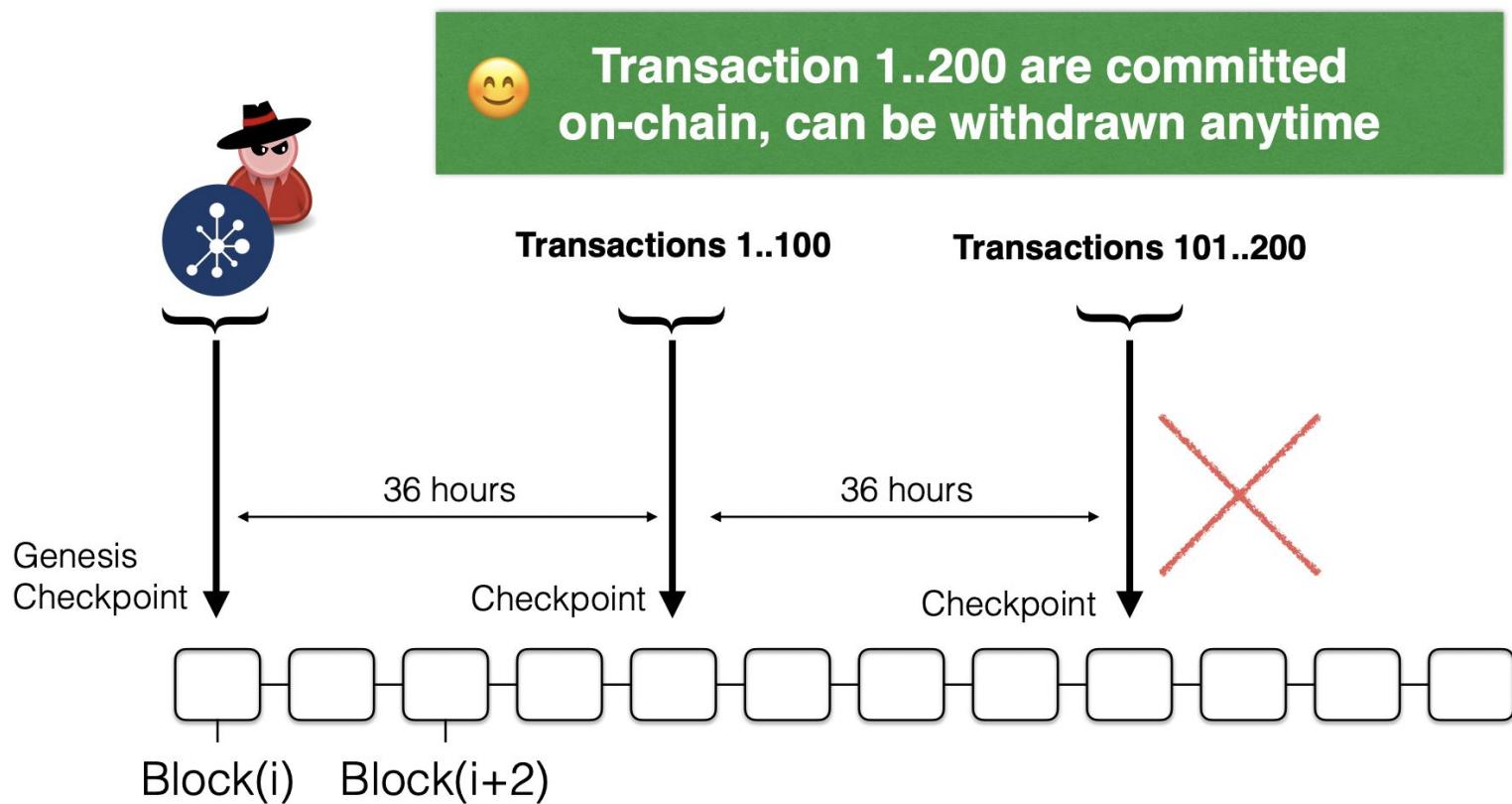
Scaling Blockchains

- NOCUST security properties



Scaling Blockchains

- NOCUST operator disappears
 - Just after checkpoint



Scaling Blockchains

- NOCUST server disappears
 - After a few transactions



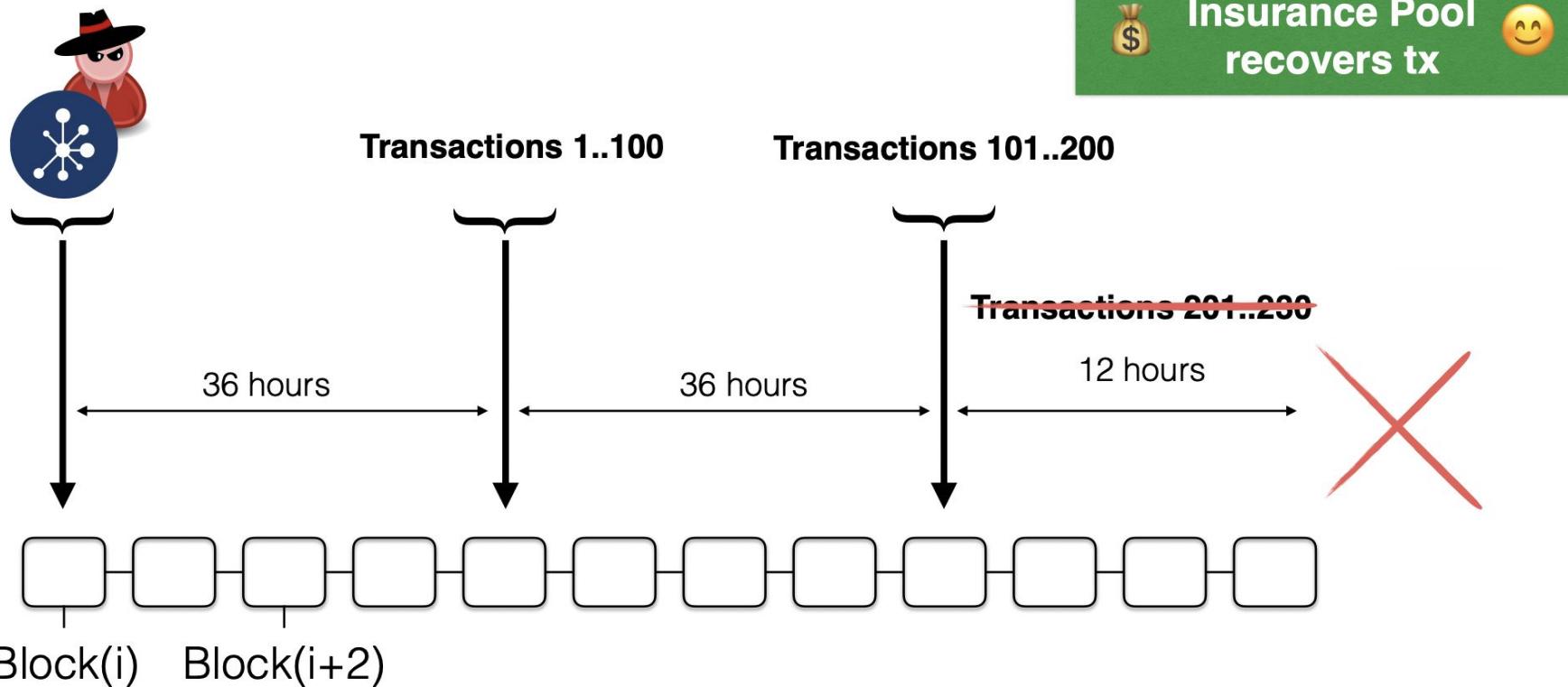
Transaction 1..200
are safe



Transaction 201..230 are
not committed on chain!

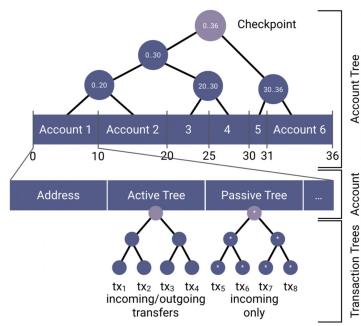
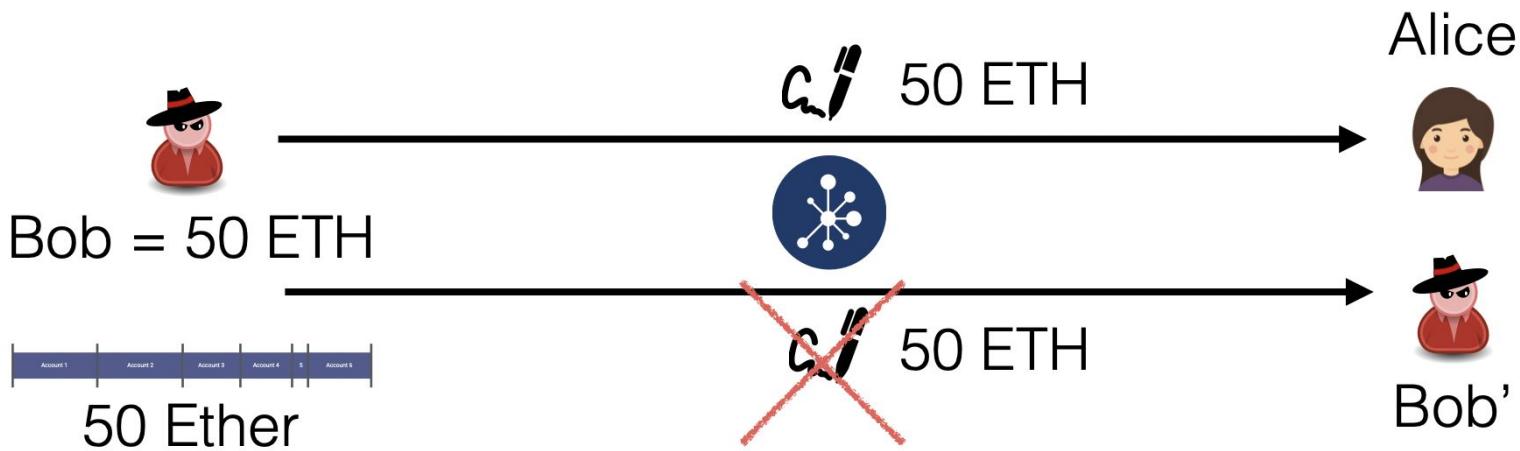


Insurance Pool
recovers tx 😊



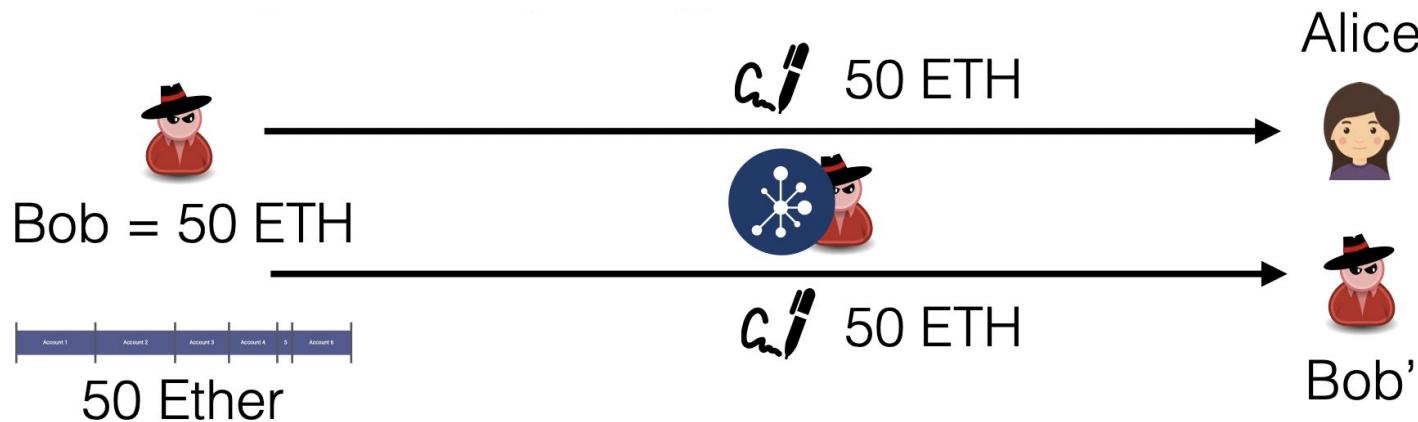
Scaling Blockchains

- Users attempt double-spending



Scaling Blockchains

- NOCUST server colludes with client to attempt double-spending



Attempt to create coins



50 Ether != 100 Ether

Operator is challenged !

Attempt to steal coins



50 Ether of Alice are lost

Operator is challenged !

Scaling Blockchains

- The good, bad, and ugly of commit-chains



Recipient can be offline to receive a transaction



Direct connection needed



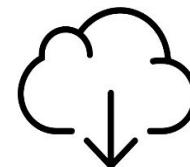
Centralized, can censor transactions



No collateral for delayed finality



Instant and free onboarding



Data availability requirements

Scaling Blockchains

- Real-world systems in production

