Blockchain

Part – II Understanding blockchain through its application: Bitcoin

Credits: Several of the slides borrowed from Prof. Ari Juels, Jacobs Technion-Cornell Institute and Prof. Aggelos Kiayias, The University of Edinburgh

It all began with Bitcoin…

2008: Breakthrough

Bitcoin: A Peer-to-Peer Electronic Cash System

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

Blockchain

In the Beginning, There Was The Coin
The Year 1 A.B. (a.k.a. 2009)

The Genesis Block

2009 : Birth of the Bitcoin

- Mankind discovers that a decentralized piece of software can embody a virtual currency
- That piece of software is the blockchain
- No central bank, no legal person, no regulation

Mining Nodes

- Validate transactions
- Add them to the block they are building
- Broadcast the completed block to other nodes

All these operations (especially many, many cryptographic computations) have a huge cost

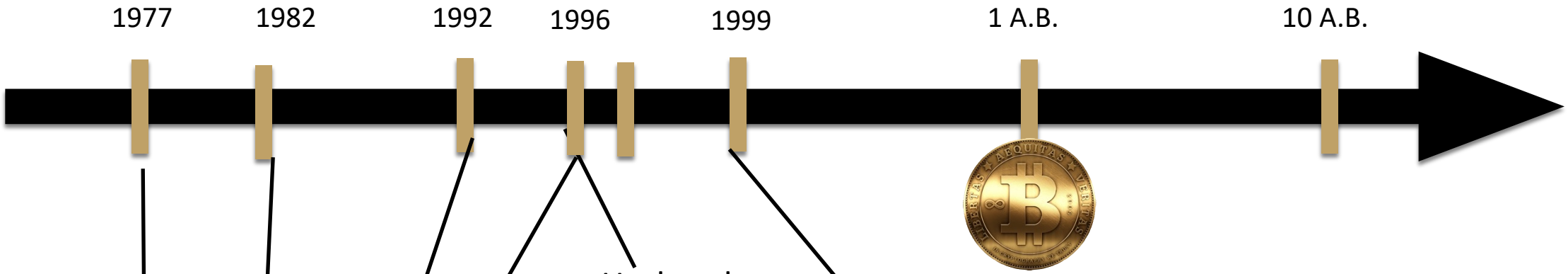

6H
E
4F

It all began with Bitcoin…

Or did it???

Actually…

32 B.B.



- RSA invented—digital currency an early proposed application
- Lamport paper on Byzantine Generals problem
- Anonymous cryptocurrency introduced (Chaum)
- Proofs of work introduced (Dwork and Naor)
- First proof-of-work-based cryptocurrency (Rivest and Shamir)
- “Proof of Work (PoW)” defined and formalized
- Hashcash

Bitcoin through the lens of… money

Two flavors of financial instrument

1. Bearer instrument
   - Physical holder of object (or secret) possesses asset
     - E.g., cash, bearer bond
   - Pros:
     - Easy to transfer
     - Anonymous
   - Cons:
     - Easy to lose!

Two flavors of financial instrument

2. Registered instrument
   - Asset with centralized record of ownership
     - E.g., most types of shares or bonds
   - Pros:
     - Can’t (physically) lose it!
   - Cons:
     - Not anonymous
     - Hard to transfer… unless register is digital…
- Both types of instrument have a venerable history

What is money?

- Resource with widespread use as medium of exchange
- Four key ingredients for bearer version, mechanisms for:
  1. Creation
  2. Forgery prevention
  3. Verification of validity / ownership
  4. Transfer between parties

Example

What are mechanisms for:

1. Creation
2. Forgery prevention
3. Verification of validity / ownership
4. Transfer between parties

????

Forgery prevention of banknotes

Watermark

A genuine banknote incorporates a watermark of Atatürk's portrait and the denomination numeral, which can be seen in the unprinted white area when the banknote is held up to the light from either the obverse or reverse side.

Counterfeit banknotes either do not contain watermarks at all or they imitate the watermark with an imprint inside or on the paper.

Original

Counterfeit

Security Fibers

Security fibers are incorporated into a genuine banknote during the production process by the Central Bank. They are invisible to the naked eye but fluoresce in blue and red under UV light.

Counterfeit banknotes attempt to imitate these security fibers through an imprint or the use of highlighters.

Original

Counterfeit

https://www.tcmb.gov.tr/wps/wcm/connect/EN/TCMB+EN/Main+Menu/Banknotes/Counterfeit+Banknotes/Characteristics+of+Counterfeit+Turkish+Lira+Banknote s

Money

- In the mid 7th century, in Lydia and Ionia (modern Turkey), the first *coins* were struck.
- For forgery prevention, coinage usually relied on three things:

A. Tokens made from scarce resource

- Electrum (gold and silver)

B. Sign / signature that was hard to duplicate

- Initially drew on skills of gem-engravers

C. Death penalty for forgers didn't hurt

- This solution lasted for many centuries…

Alyattes Trite (Lydia 1/3 stater).
Image Courtesy of CNG: www.cngcoins.com.

Intaglio depicting goddess Demeter. 1st cent. B.C.E. Private collection.

2600+ years later…

Still uses classic principles!

For forgery prevention:

A. Scarce resource: computation
B. Hard-to-forge data: cryptography

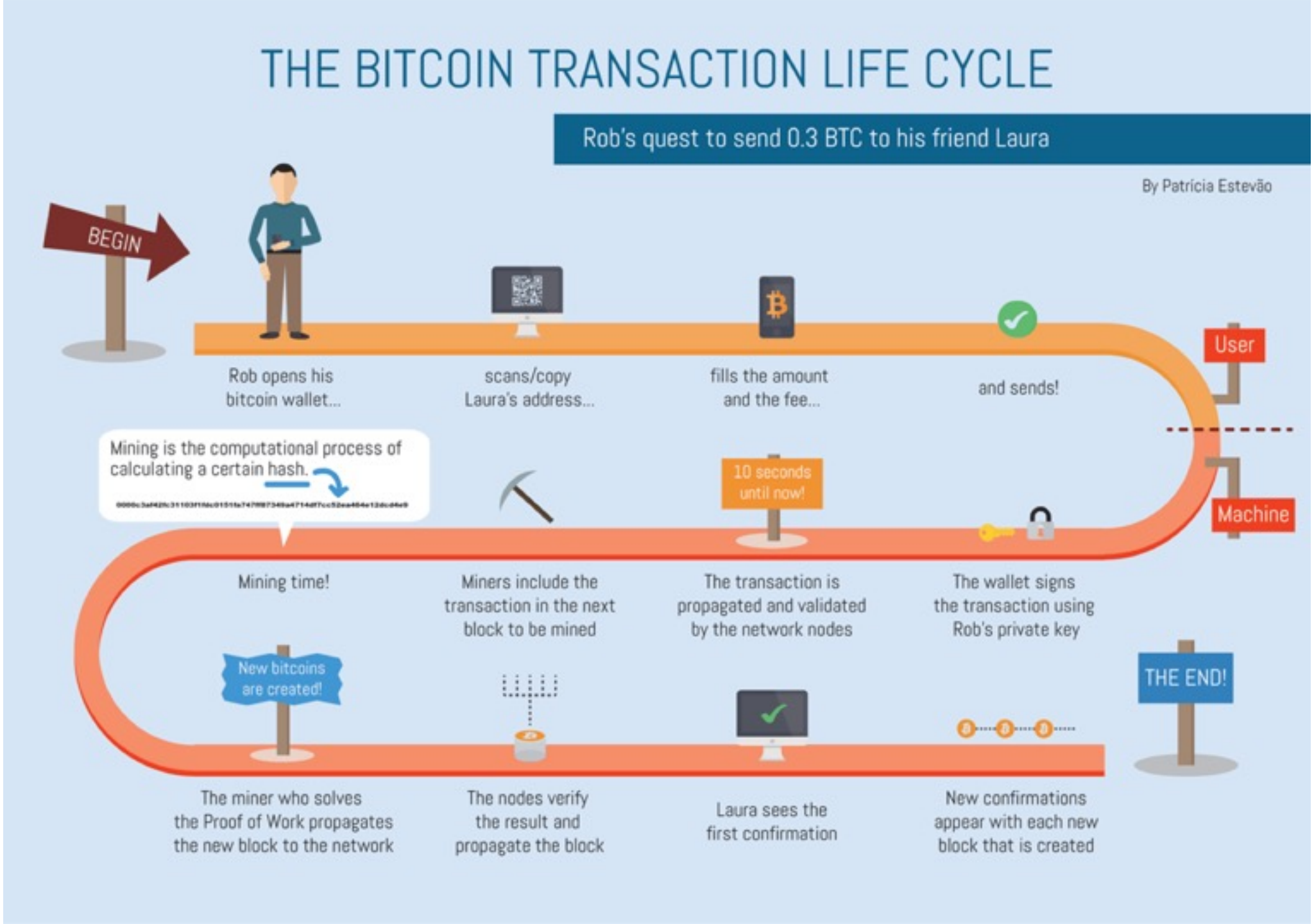
Bitcoin

What kind of money is Bitcoin?

- Combines:
  - A ledger: Public record of asset ownership called the *blockchain*
  - Secret keys for control of individual's assets
- So it’s got features of bearer instruments *and* registered instruments
- Quasi-anonymous (pseudonymous)
- Loose comparisons to Yap currency, etc., but no perfect physical analog

“Anyone got change?”

Blockchains and Bitcoin: the Bird’s Eye View

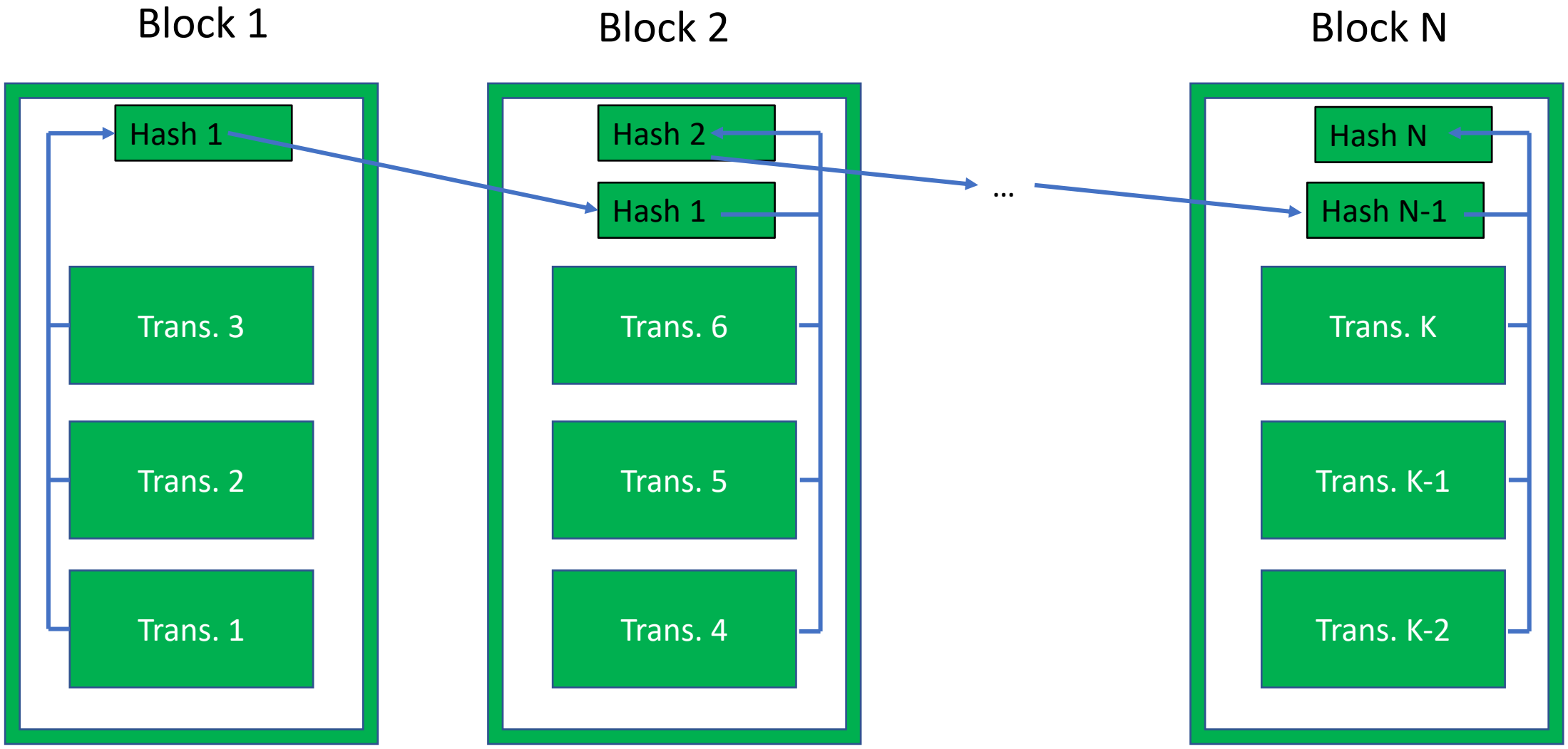

THE BITCOIN TRANSACTION LIFE CYCLE
Rob's quest to send 0.3 BTC to his friend Laura
By Patrícia Estevão
BEGIN
Rob opens his bitcoin wallet...
scans/copy Laura's address...
fills the amount and the fee...
and sends!
User
Machine
Mining is the computational process of calculating a certain hash.
10 seconds until now!
Mining time!
Miners include the transaction in the next block to be mined
The transaction is propagated and validated by the network nodes
The wallet signs the transaction using Rob's private key
New bitcoins are created!
THE END!
The miner who solves the Proof of Work propagates the new block to the network
The nodes verify the result and propagate the block
Laura sees the first confirmation
New confirmations appear with each new block that is created

Transactions

	Sender	Receiver	Amount
Trans. 1	Alice	Bob	10 Francs
Trans. 2	Charles	Deborah	25 Francs
Trans. 3	Deborah	Emil	180 Francs
Trans. 4	Alice	Francis	20 Francs

A crucial challenge: prevent double spending!

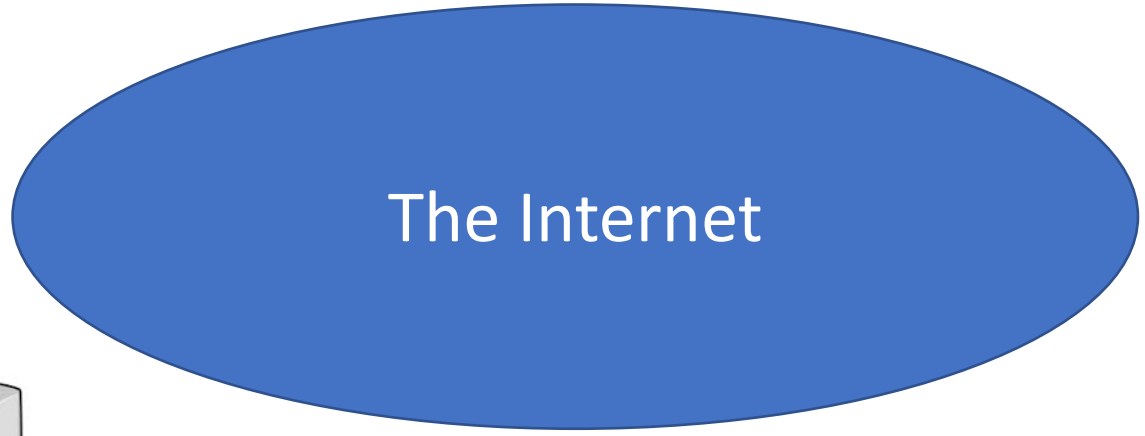
The Blockchain: Immutable Ledger of Transactions

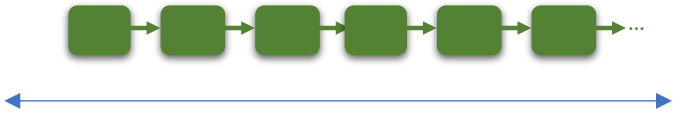


Hash: One-way hash function: Unique fingerprint of its inputs

Decentralization: Replication of the Blockchain

Along with the hash values, replication guarantees unforgeability of the blockchain



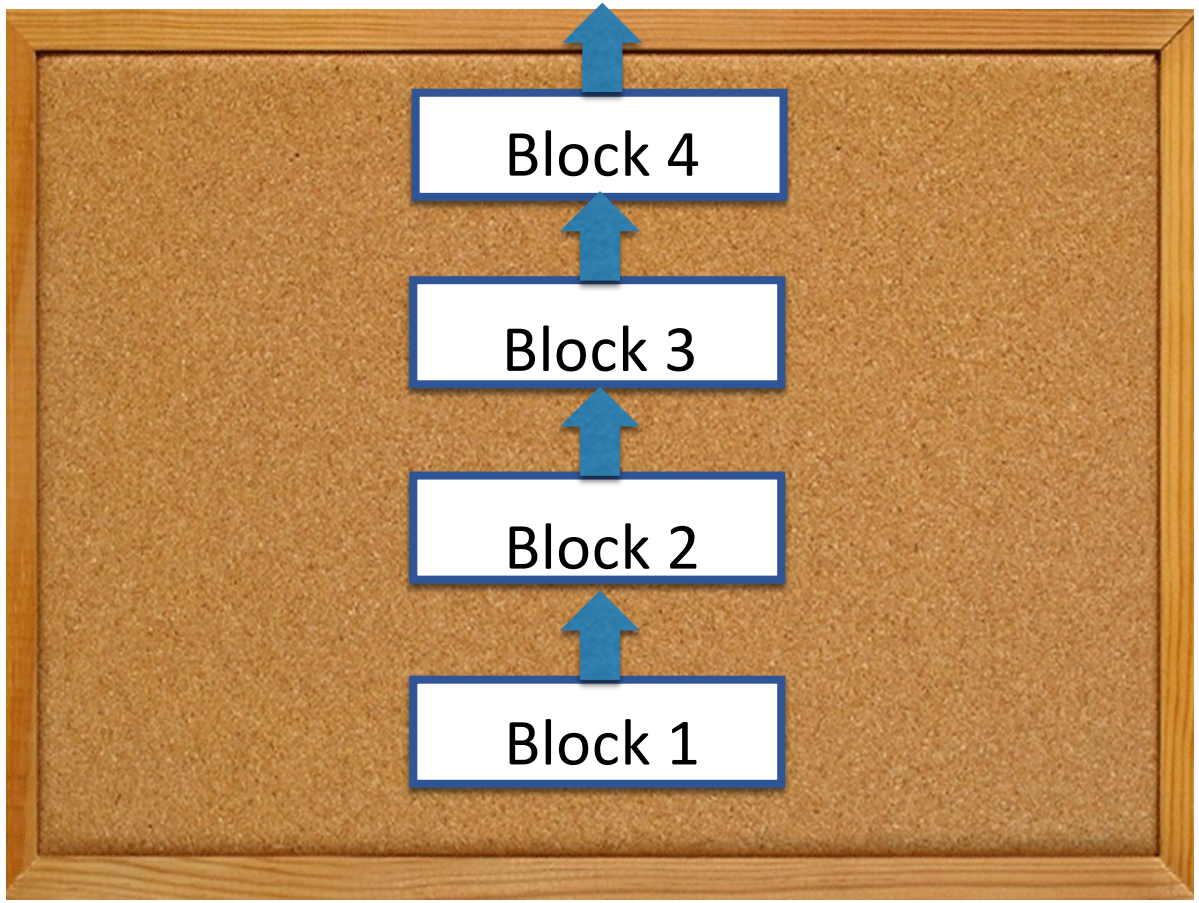


Around 200 Gbytes in the case of Bitcoin

Blockchains: Abstraction

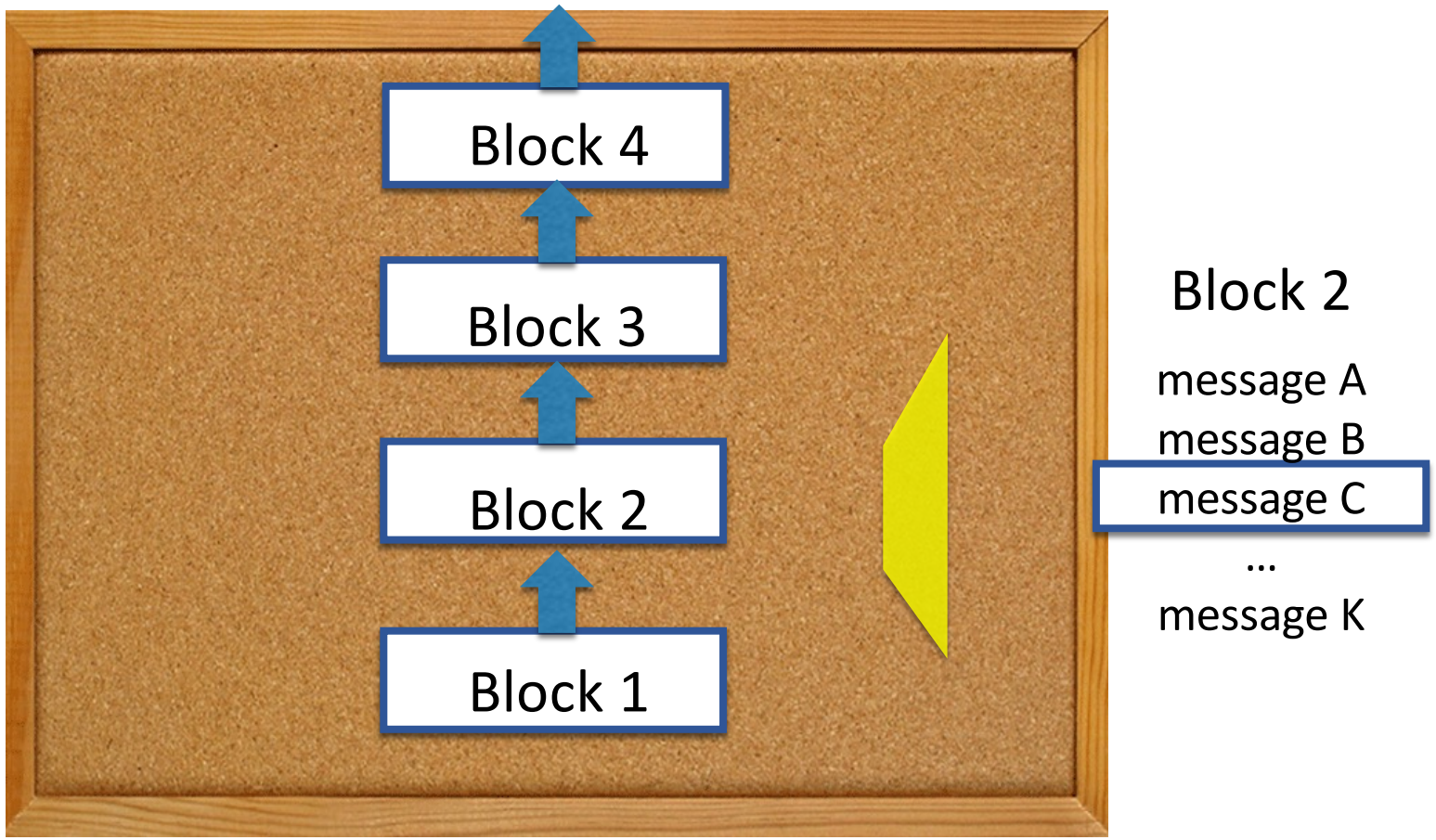
- What is the technology behind this incredible and incredibly varied promise? -> **a bulletin board**

Blockchains: Abstraction



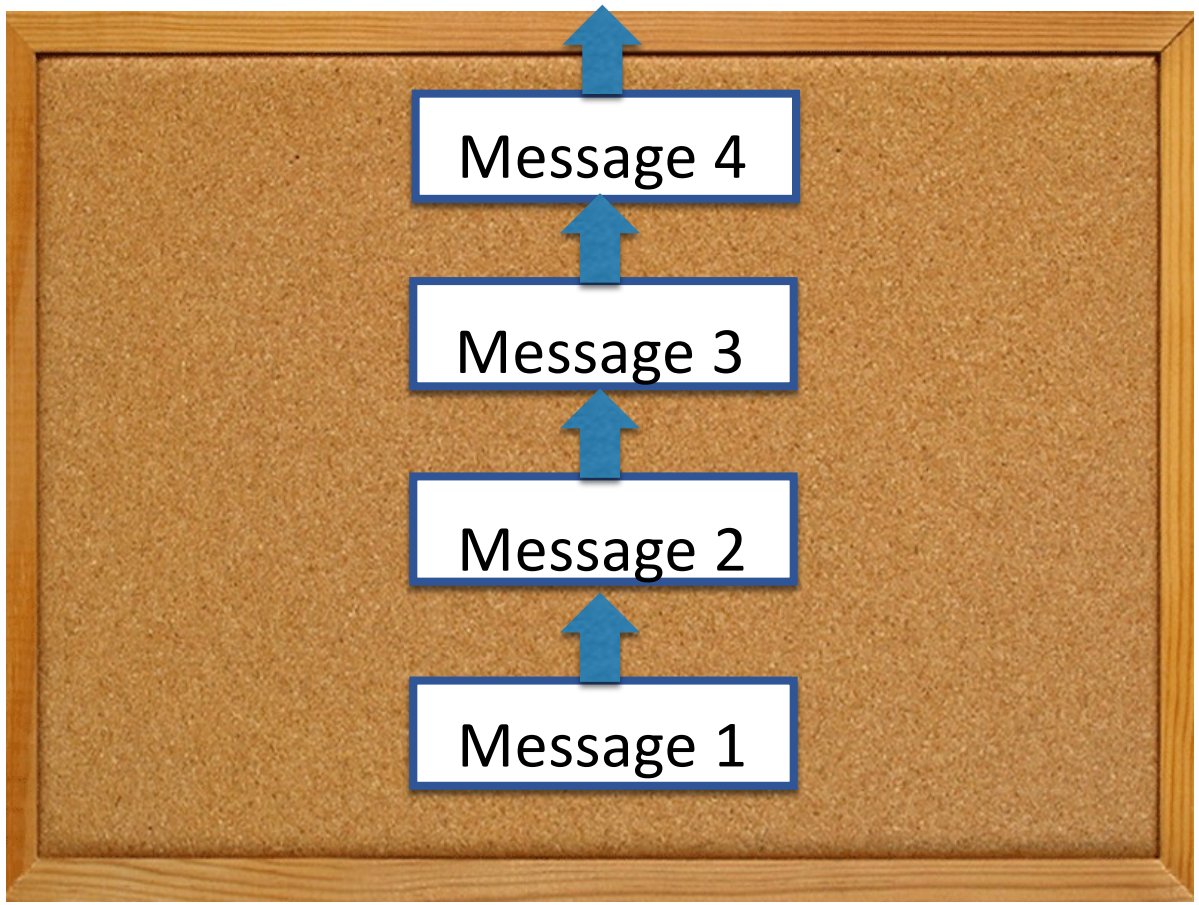
Blockchains: Abstraction

#1 Strict ordering of messages



Blockchains: Abstraction

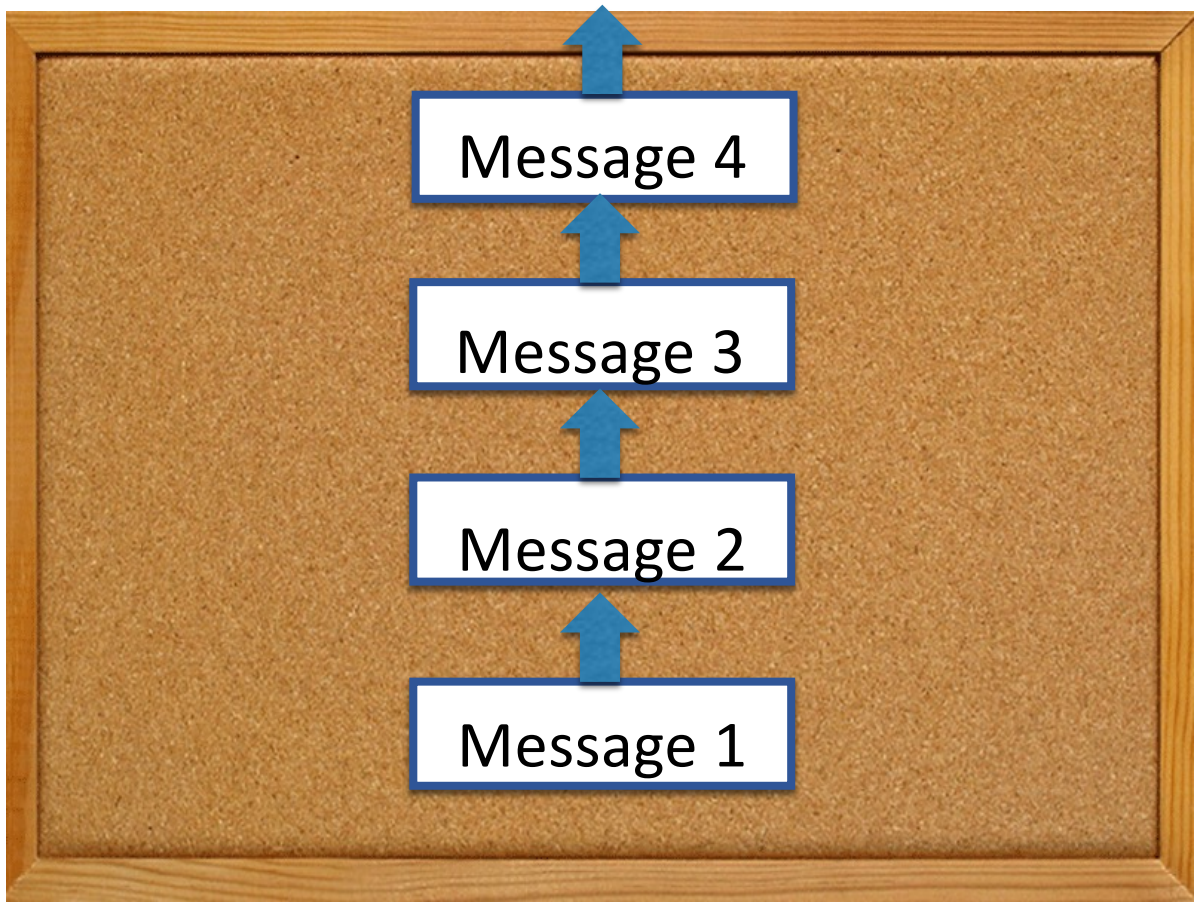
#1 Strict ordering of messages



Blockchains: Abstraction

#2 Rule-based write, global read

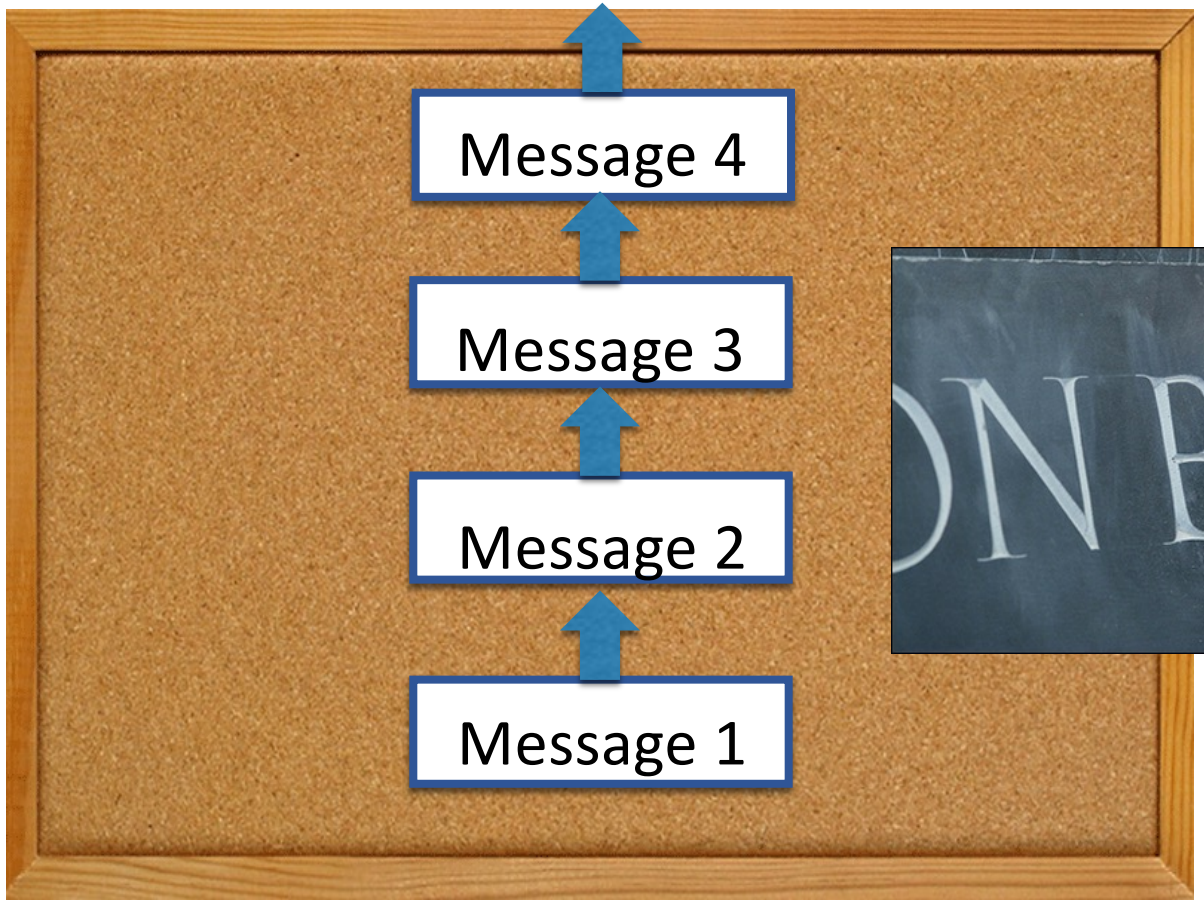
Write Permission: Rule-based



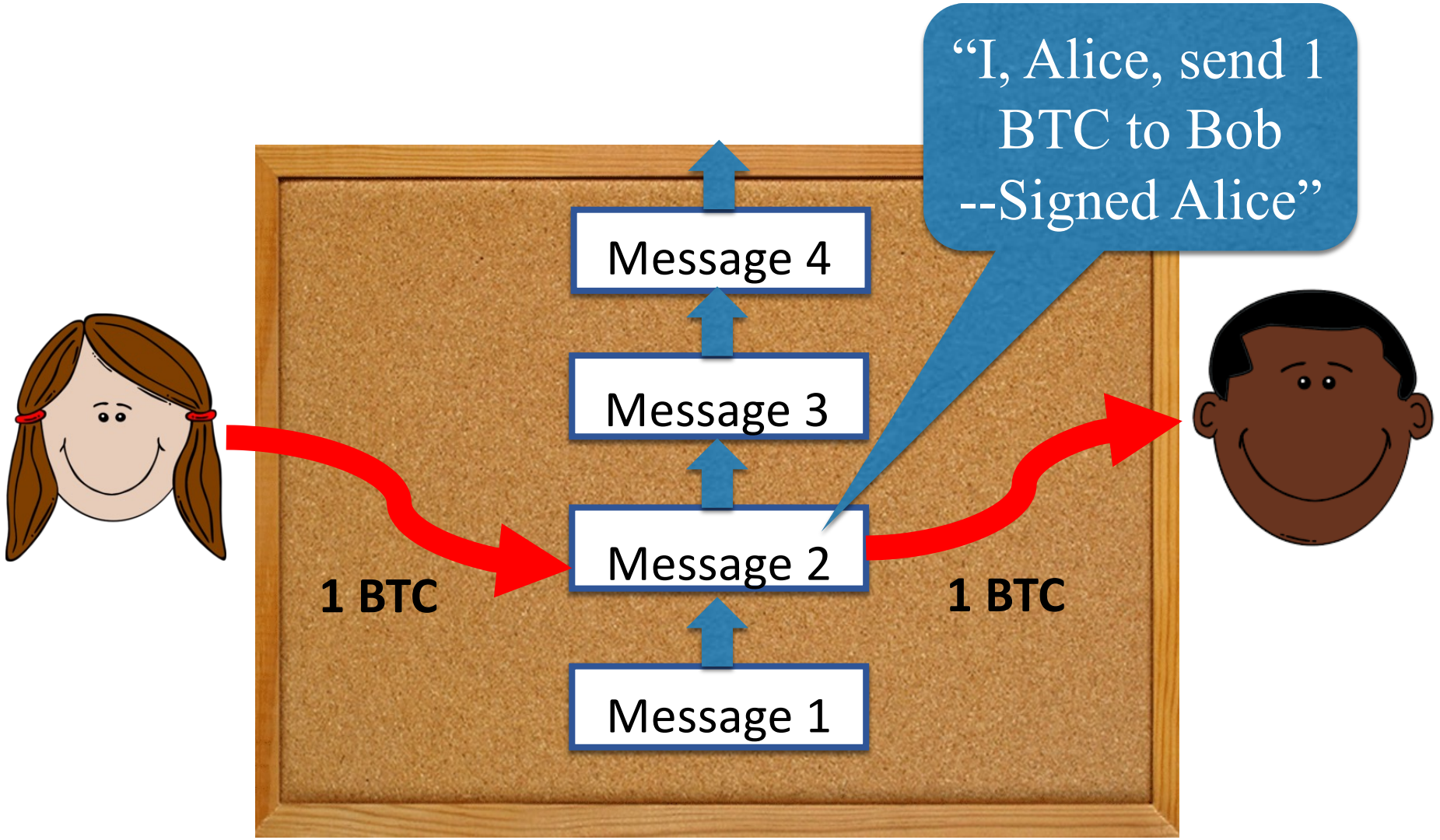
Read Permission:

Blockchains: Abstraction

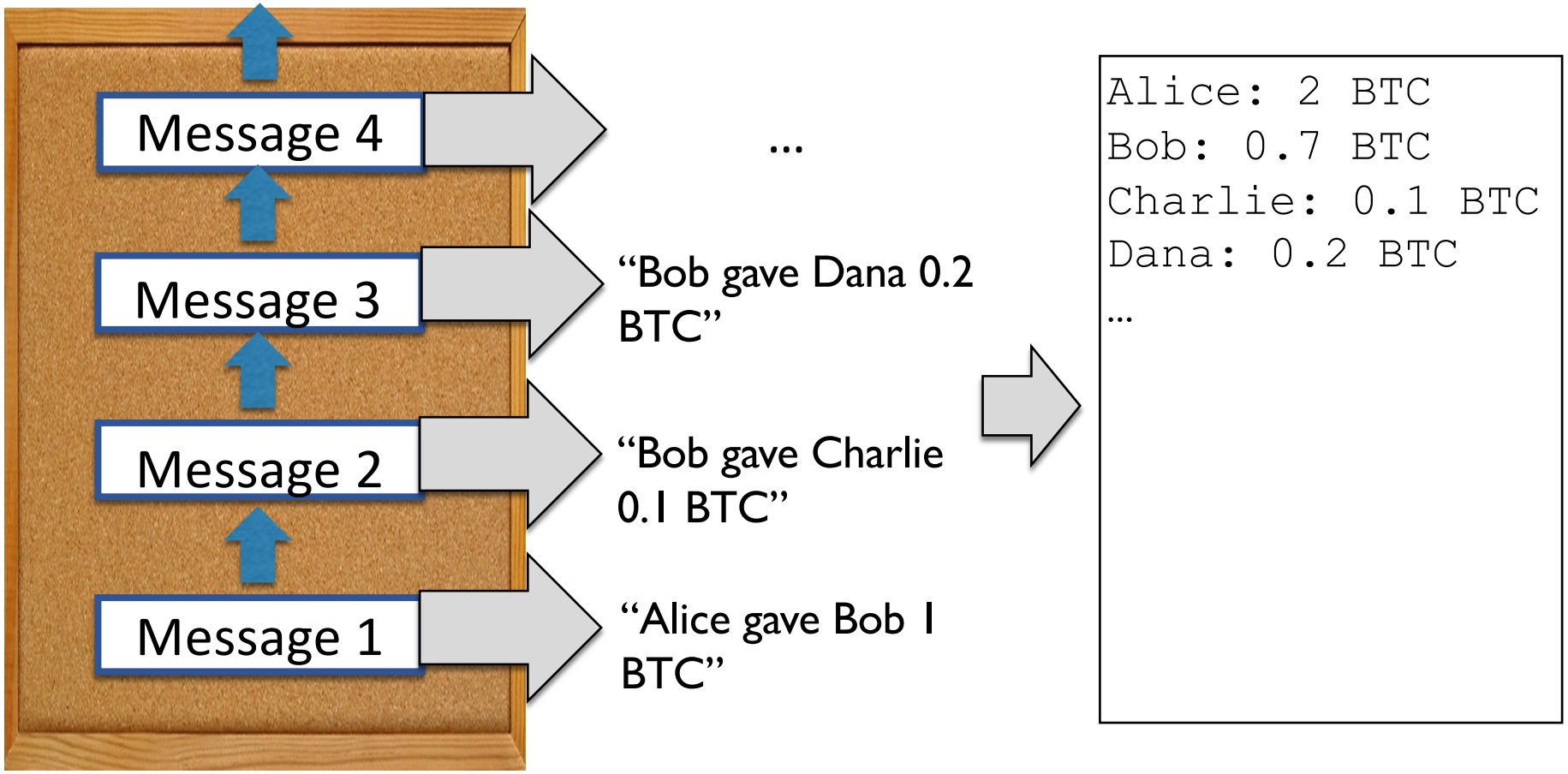
#3 No message modification



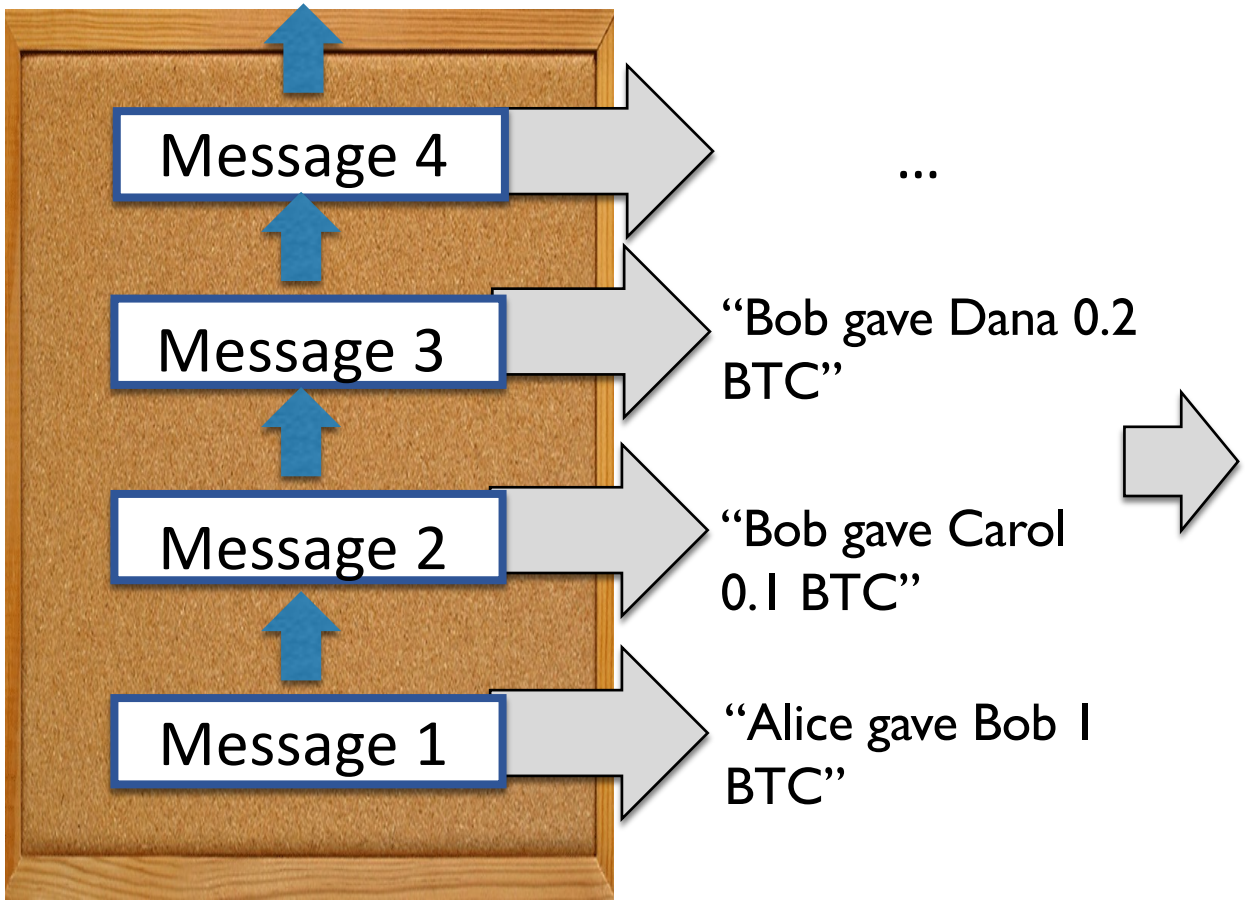
Power of the Abstraction

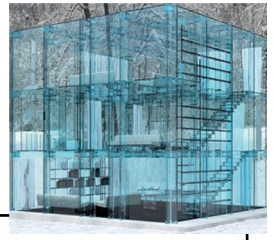


Power of the Abstraction



Blockchains are transparent…

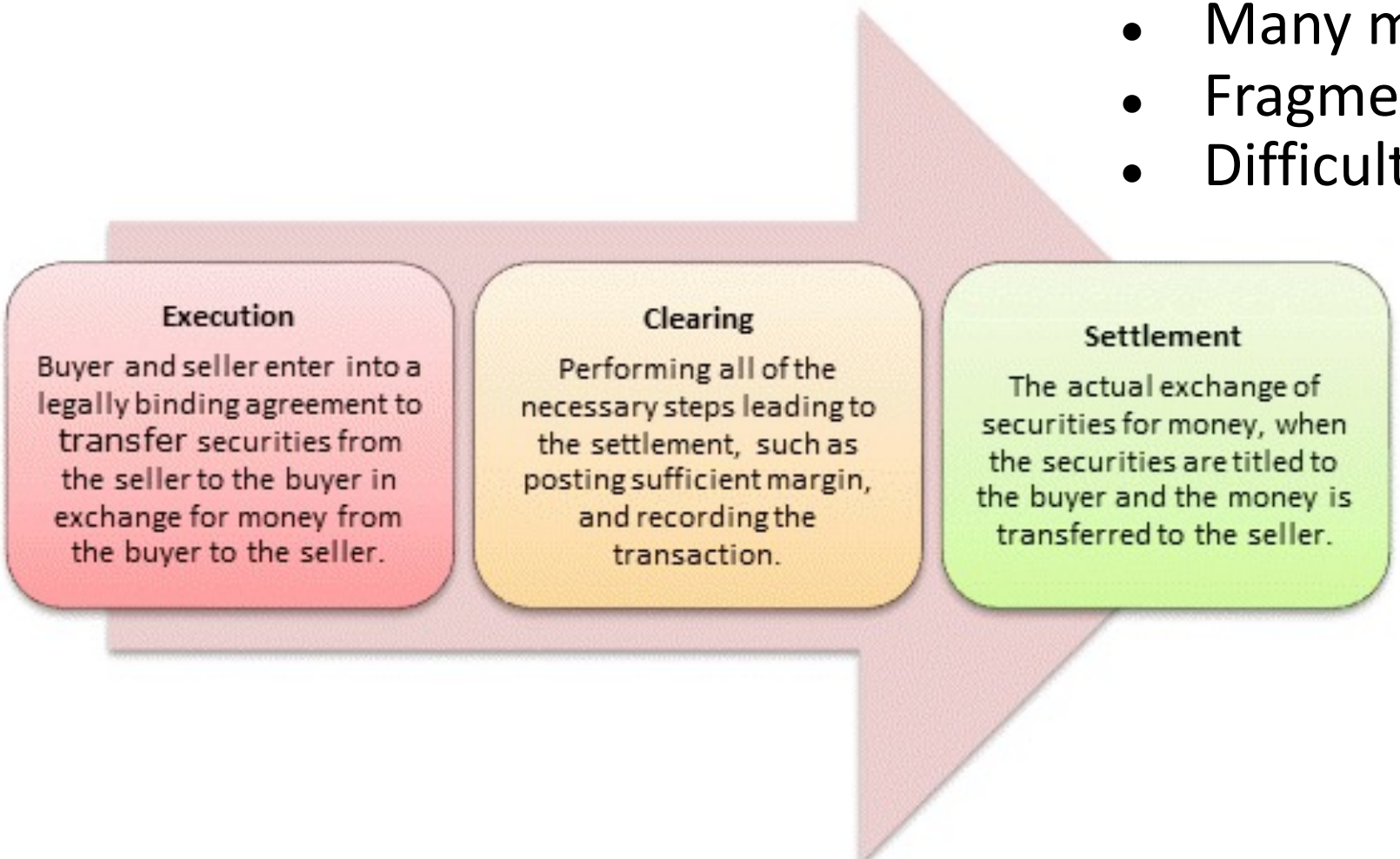




Alice: 2 BTC
Bob: 0.7 BTC
Carol: 0.1 BTC
Dana: 0.2 BTC
…

Compare: Execution, clearing, and settlement

- For transfer of financial instruments
- Up to three days to complete (T+3)
- Many middlemen
- Fragmented records
- Difficult to audit



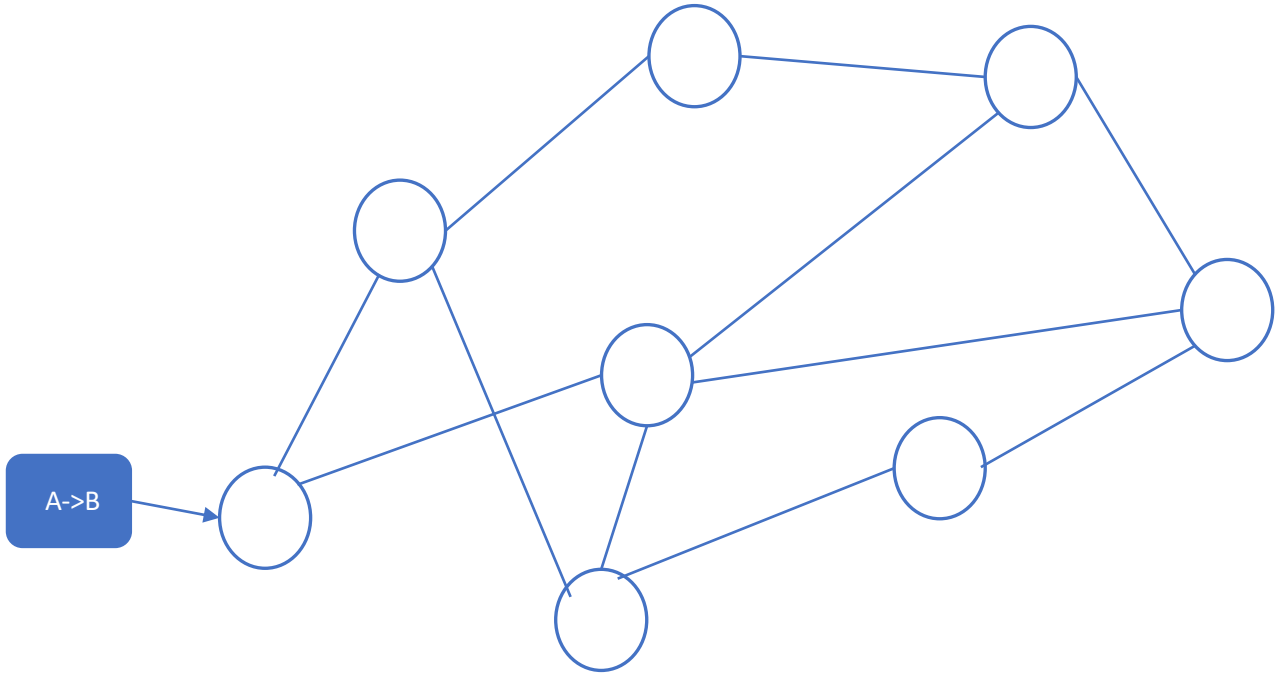
Motivation for the Consensus

- A new transaction history needs to be agreed by all participants
- Participants may have diverging interests in terms of the history of transactions
- Each participant may be interested in adding a different block

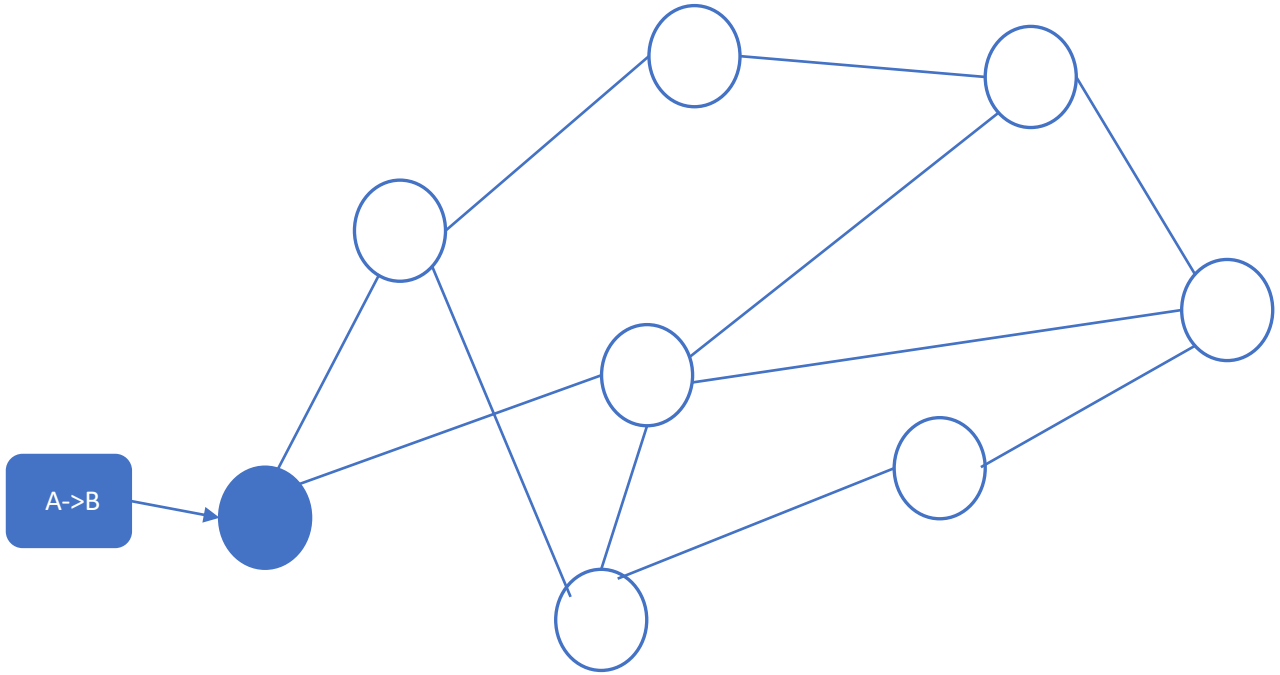
Consensus Properties

- **Termination**
  - Every correct process will eventually decide on some output
- **Integrity**
  - If all the correct processes proposed the same value v, then any correct process must decide v.
- **Agreement**
  - Every correct process must agree on the same value.

Transaction Verification in Bitcoin

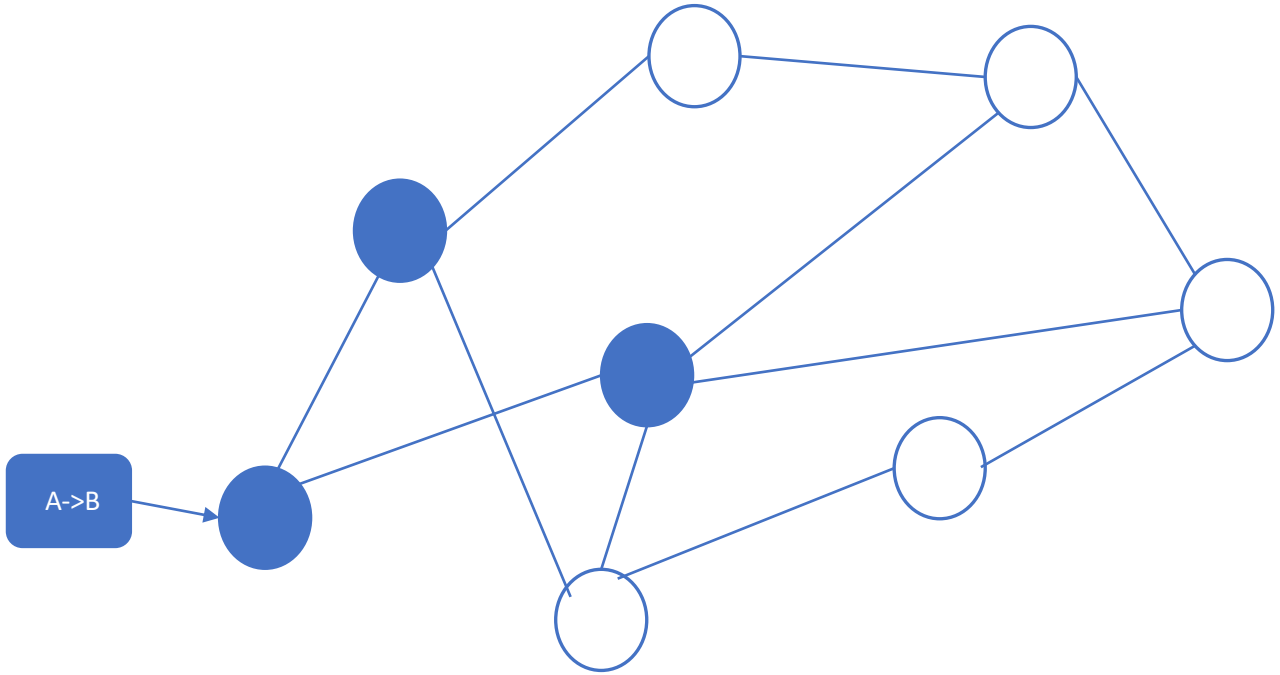


Transaction Verification in Bitcoin



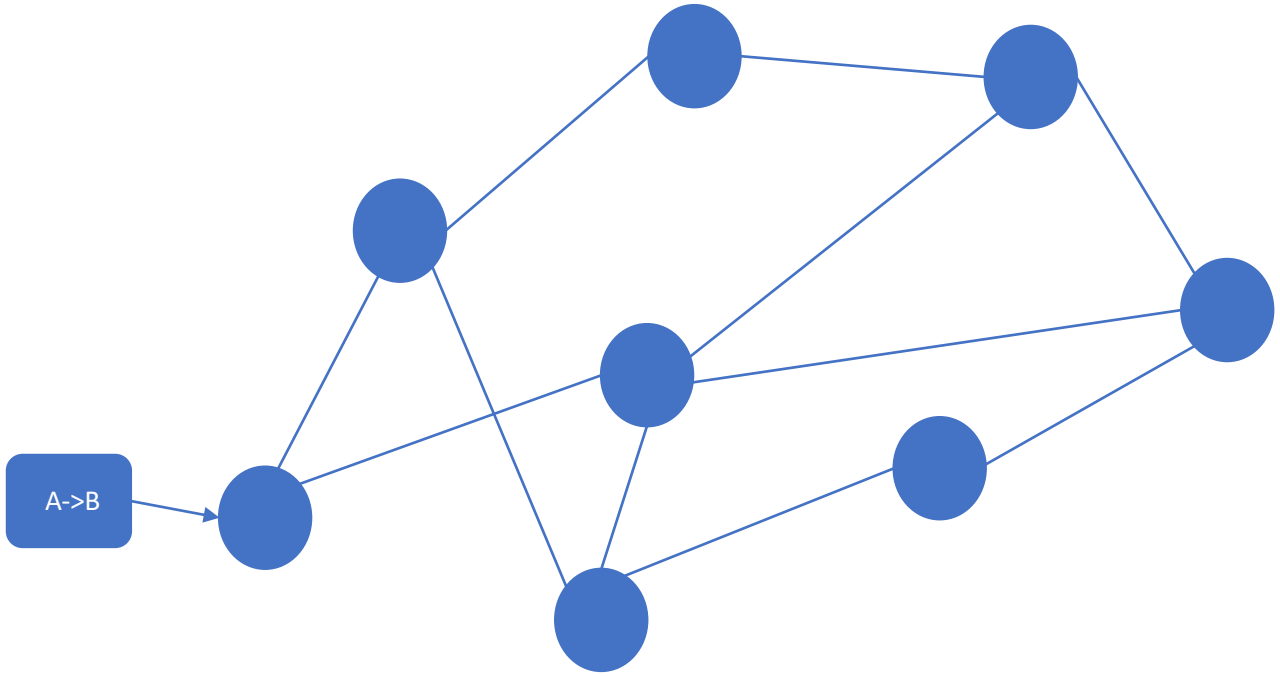
A “miner” collects a transaction from the network and propagates it to his neighbours

Transaction Verification in Bitcoin

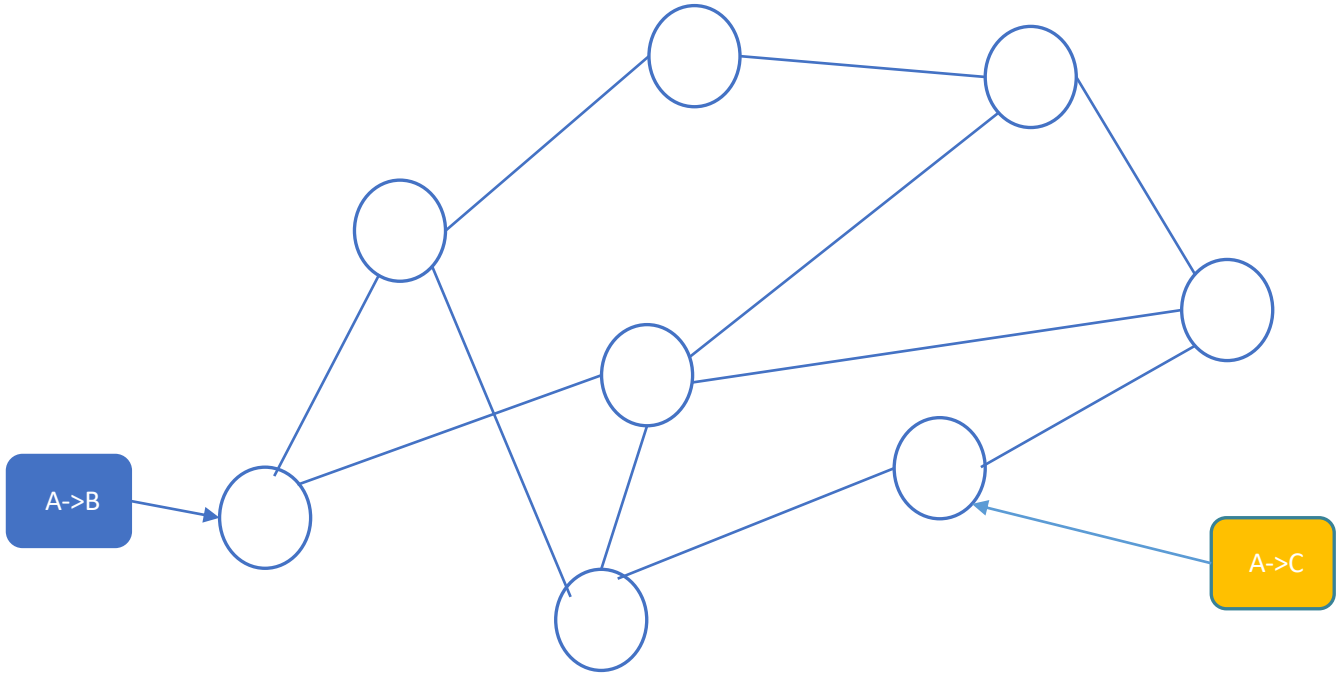


As a result everyone knows about the new transaction and updates the state

Transaction Verification in Bitcoin

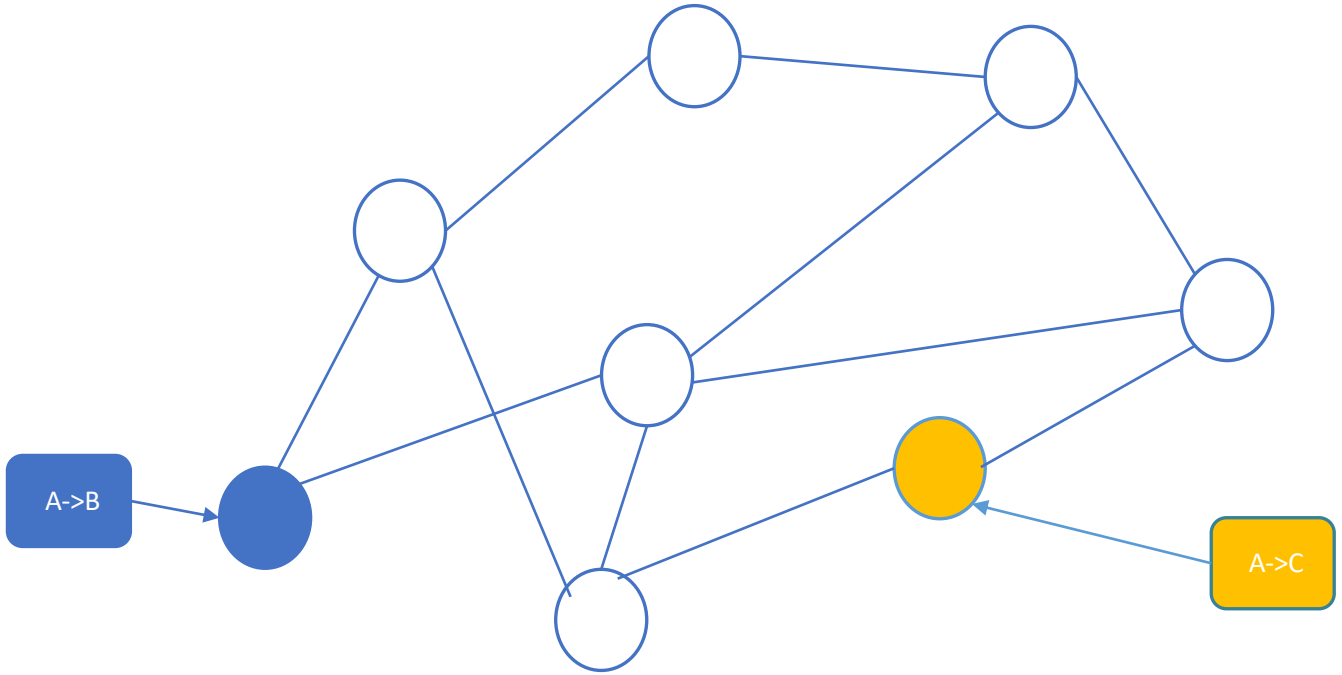


Conflict Resolution



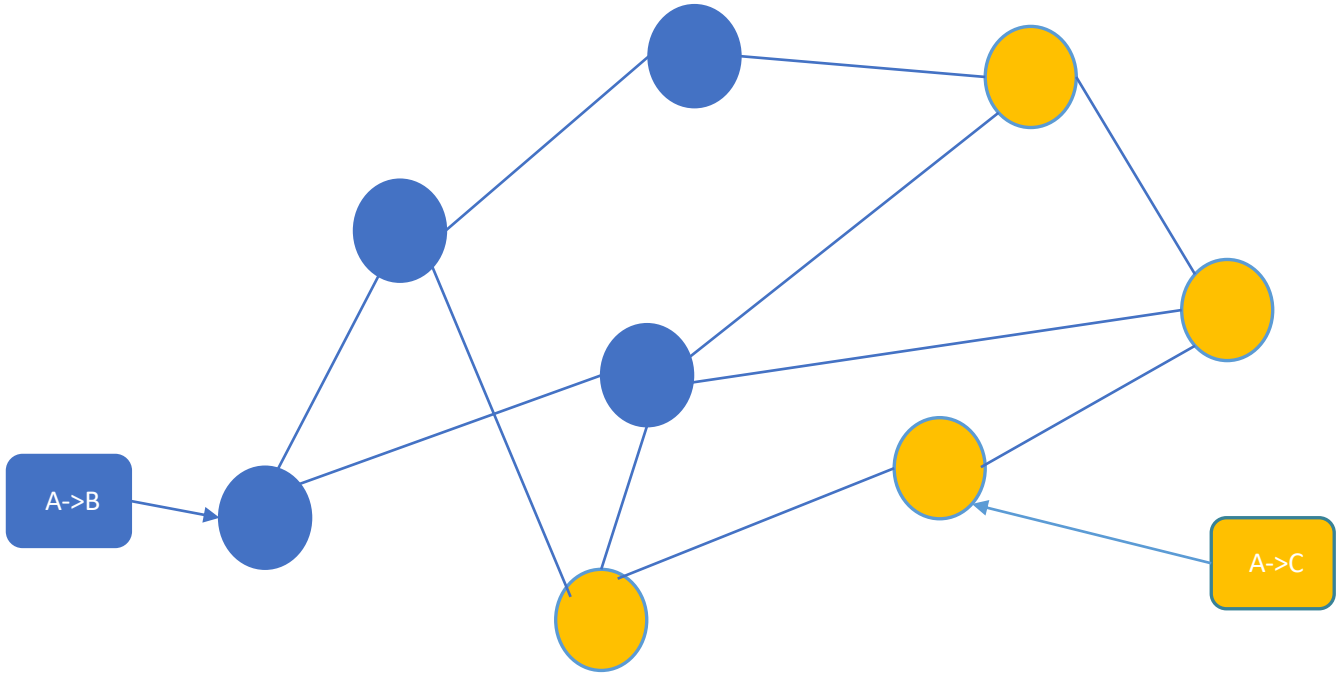
A problem if two different transactions try to spend the same funds concurrently

Conflict Resolution

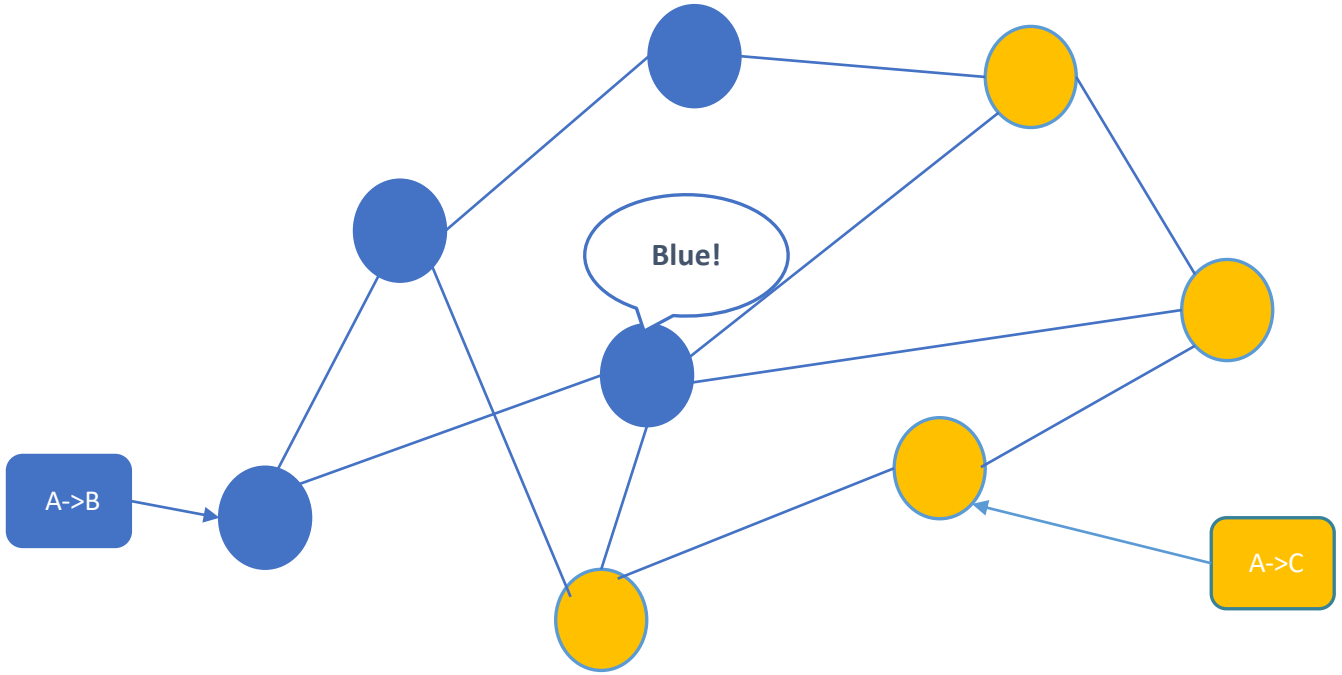


Then every node will acknowledge the transaction that is first transmitted to him and drop the other, leading to an inconsistent “bank”

Conflict Resolution

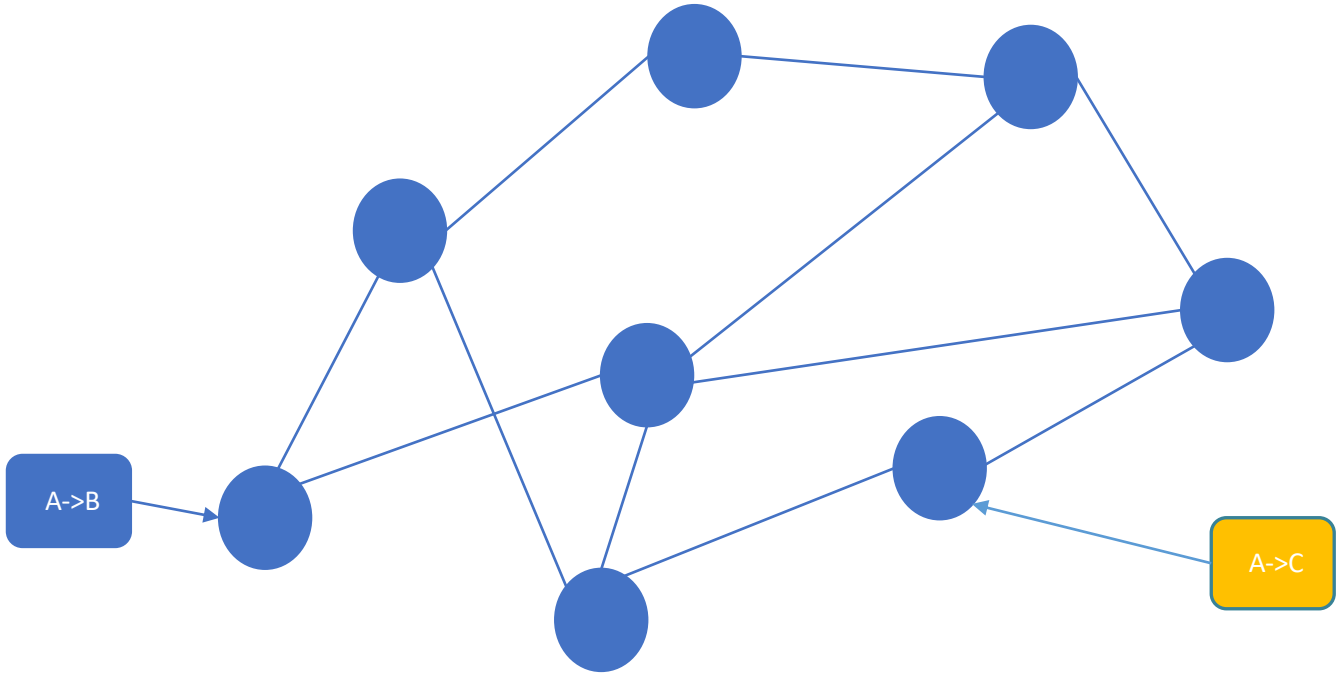


Conflict Resolution



Bitcoin, solves this problem by having a “leader” elected every 10 minutes that states which transactions are valid.

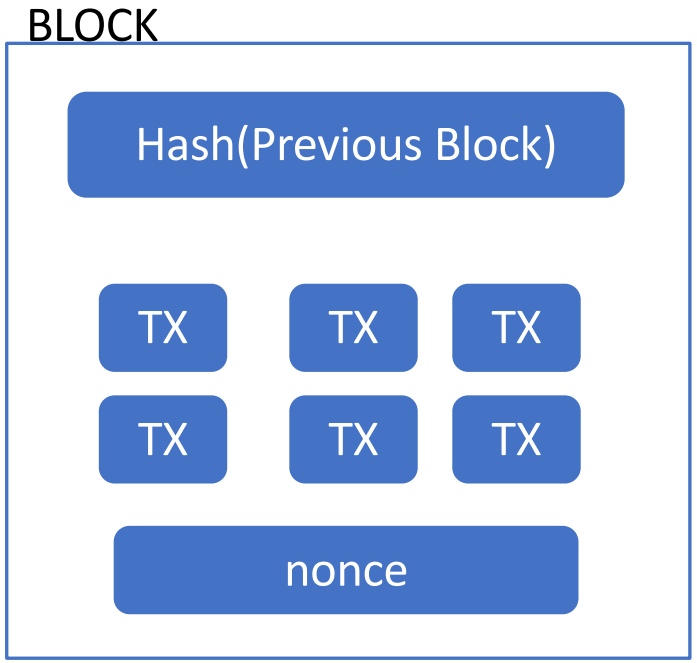
Conflict Resolution



Leader Election

How exactly is this leader elected?

Proof-of-Work



H(Block, nonce=0) =abc3426fe31233

H(Block, nonce=1) =fe541200abc229

H(Block, nonce=2) =0bc3429831233

.

.

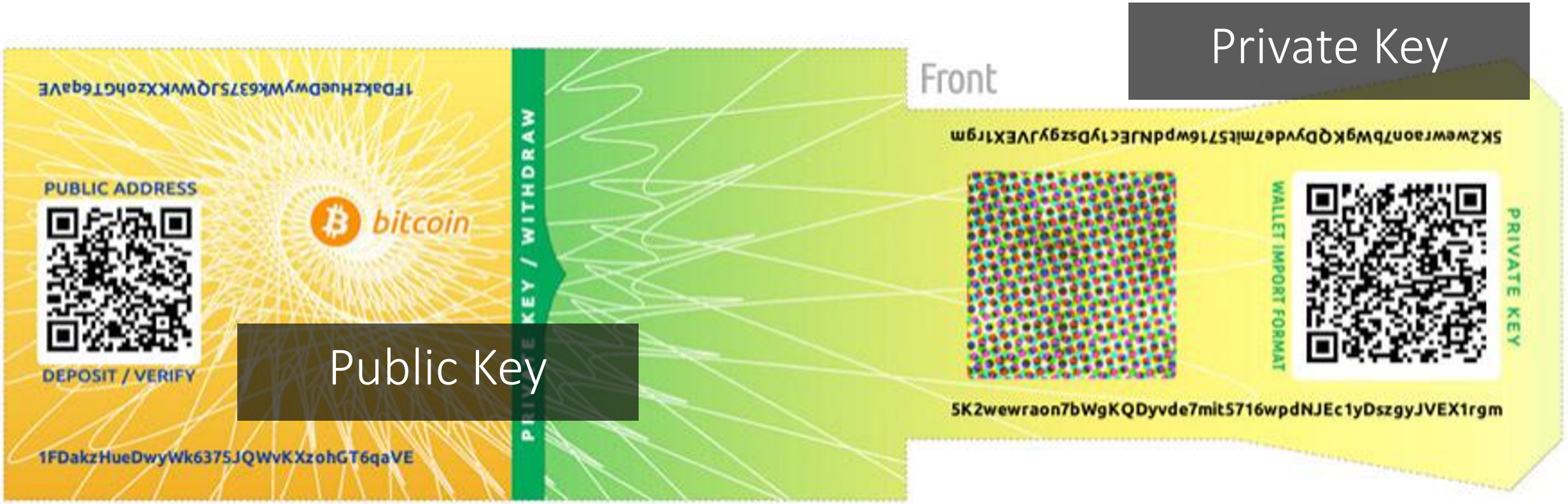
.

.

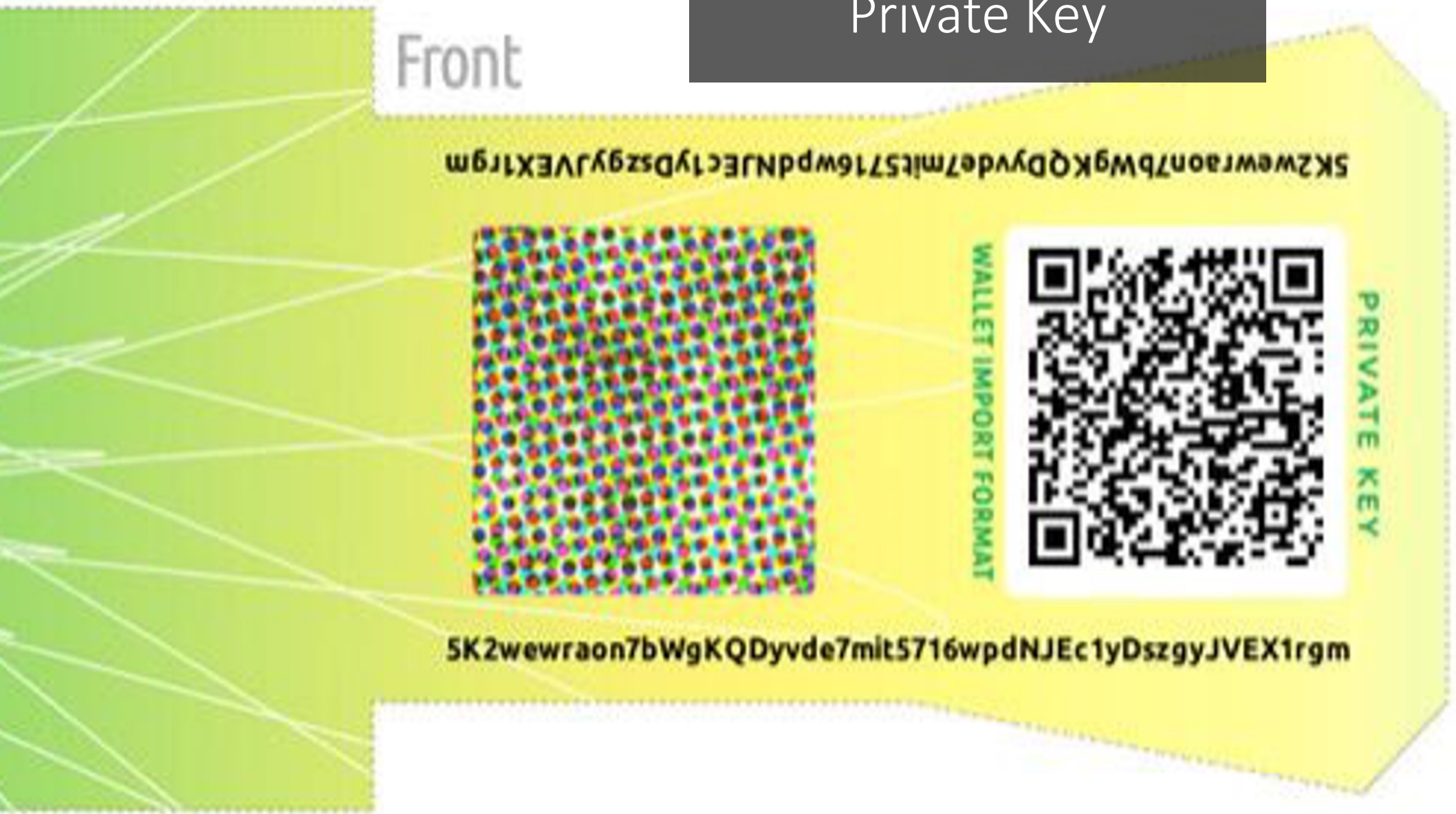
H(Block, nonce=f23) =0000fed98312

To become the leader, the nodes need to solve a puzzle. This puzzle is called **proof-of-work**

Bitcoin Paper Wallet



Private Key

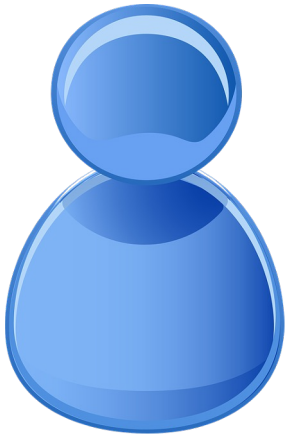

Front
5K2wewraon7bWgKQDyvde7mit5716wpdNJEc1yDszgyJVEX1rgm
WALLET IMPORT FORMAT
PRIVATE KEY
5K2wewraon7bWgKQDyvde7mit5716wpdNJEc1yDszgyJVEX1rgm


1FDakzHueDwyWk6375JQWvKXzohGT6qaVE
PUBLIC ADDRESS
bitcoin
DEPOSIT / VERIFY
1FDakzHueDwyWk6375JQWvKXzohGT6qaVE

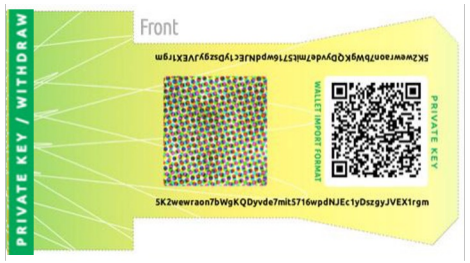
Public Key

Alice and Bob are only identified by public keys

Transfer 10 Bitcoins from me to Bob.



Alice



Signed with Alice's private key

Bitcoin Network

1BTC

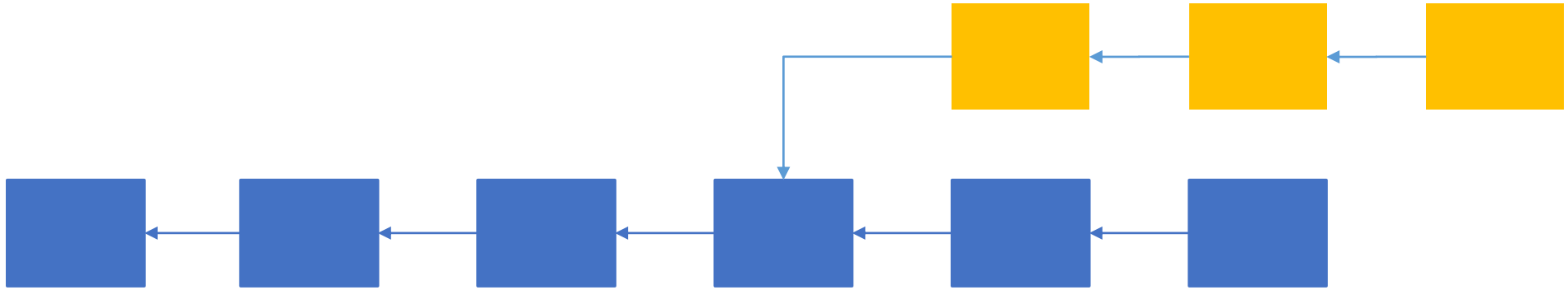
3BTC

1BTC

5BTC

Attacks on Blockchain

Unstable Consensus (Forks)



Question?

What happens if there is a network partition

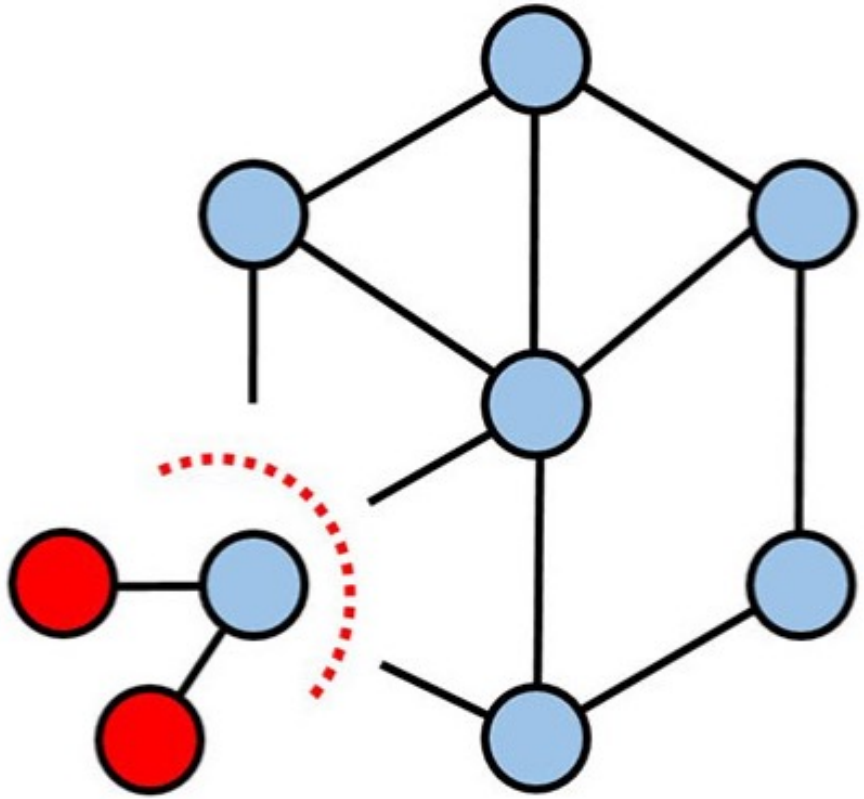
a) The protocol halts preserving safety
b) Now we have 2 versions of bitcoin that will never merge back
c) The clients do not realize it and can be attacked
d) Free money for everyone

Some well-known attack

- Eclipse Attacks -> targeting specific node
- Sybil Attacks -> targeting whole network
- Double-Spending Attacks
  - 51% Attack
  - Race Attack
- …

Eclipse Attacks

- Adversary targets a **specific node** to cut off all of its communications with other peers and isolate this specific node
- A successful Eclipse Attack enables isolating the victim node and prevent the victim from attaining a true picture of real network activity and the current ledger state
- By isolating a lot of nodes, attacker can remove significant amount of “hash power” from the system



https://www.radixdlt.com/post/what-is-an-eclipse-attack/

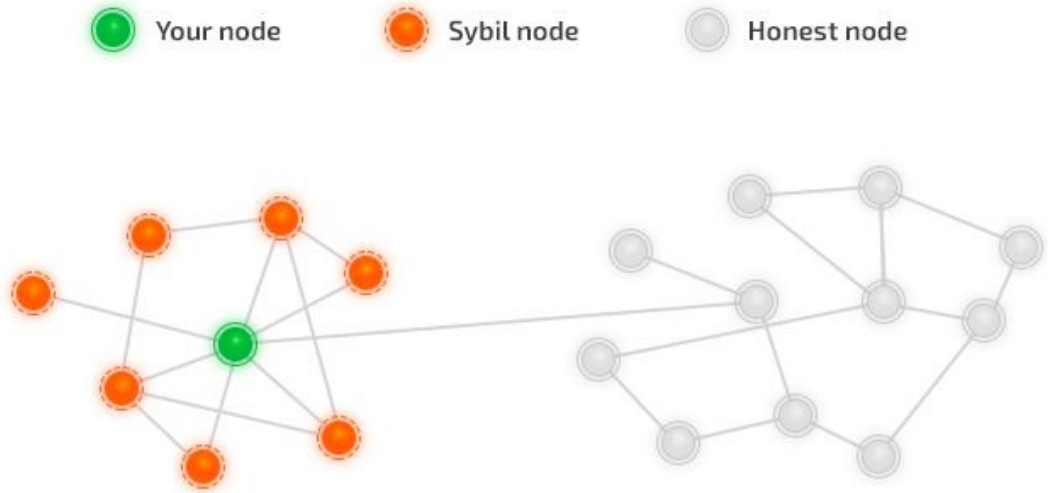
Eclipse Attacks - How to mitigate?

- Random node selection
- Fewer nodes per IP address or machine
- Information storage (storing information about nodes)
- Increasing number of connections

https://www.radixdlt.com/post/what-is-an-eclipse-attack/

Sybil Attacks

- Sybil Attack is a type of attack seen in peer-to-peer networks in which a node in the network operates multiple identities.
- The name Sybil comes from a woman named Sybil Dorsett, who had Dissociative Identity (Multiple Personality) Disorder



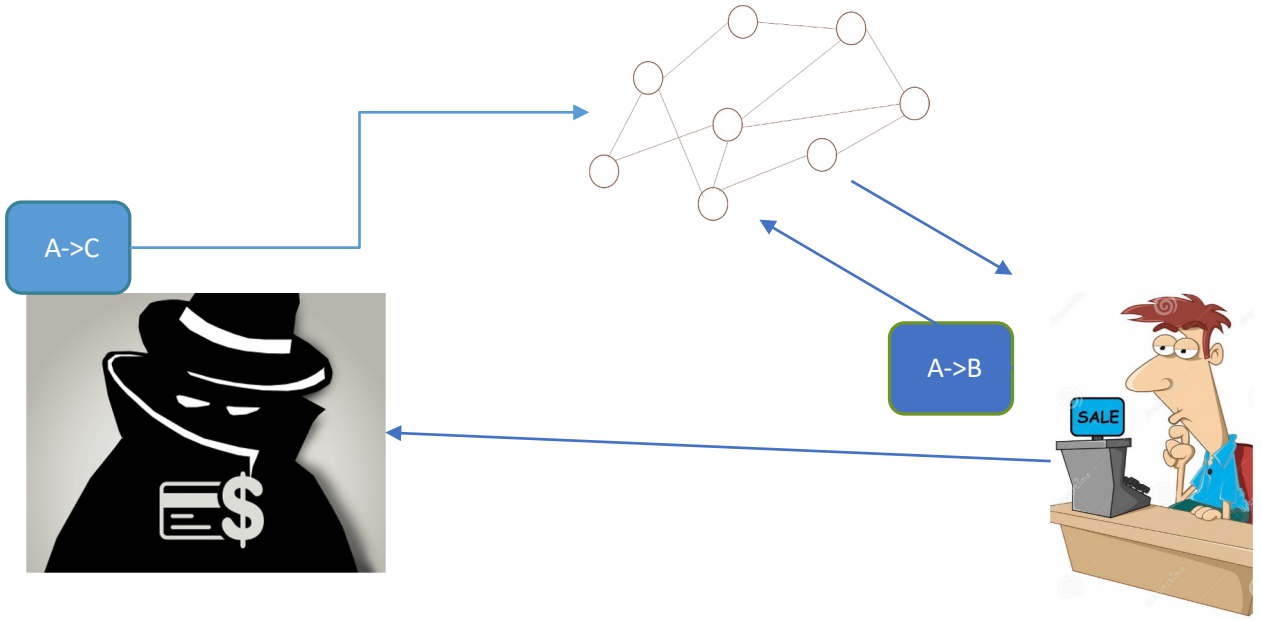
- **Example:** The 51% attack in Blockchain networks

Sybil Attacks - How to mitigate?

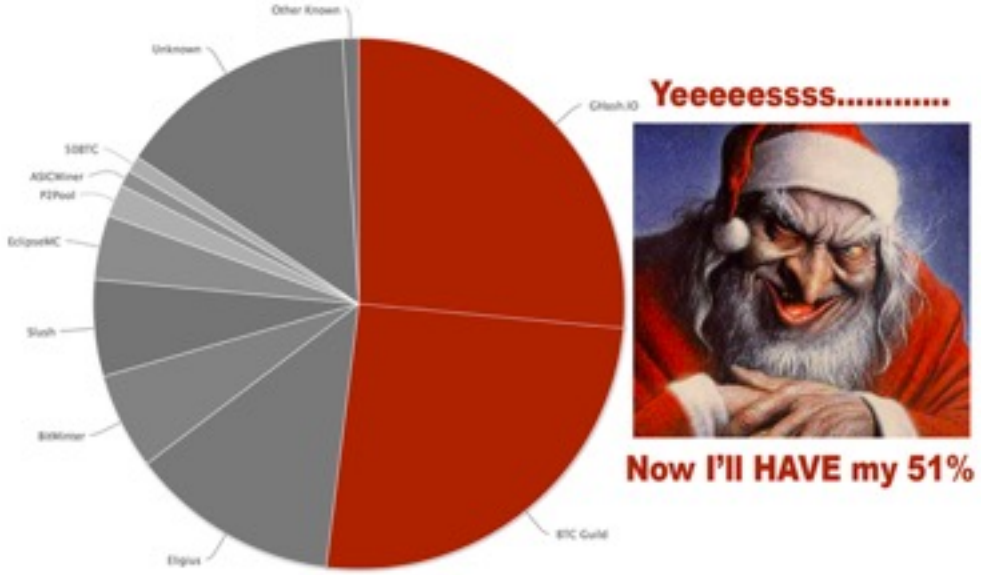
- Blockchains use different consensus algorithms
- Bitcoin uses Proof of Work (PoW) consensus algorithm to prove the authenticity of any block that is added to the blockchain

Double Spending Attacks

1) Give transaction to seller
2) Take the product
3) Send a 2nd transaction and create a longer chain



Double Spending Attacks

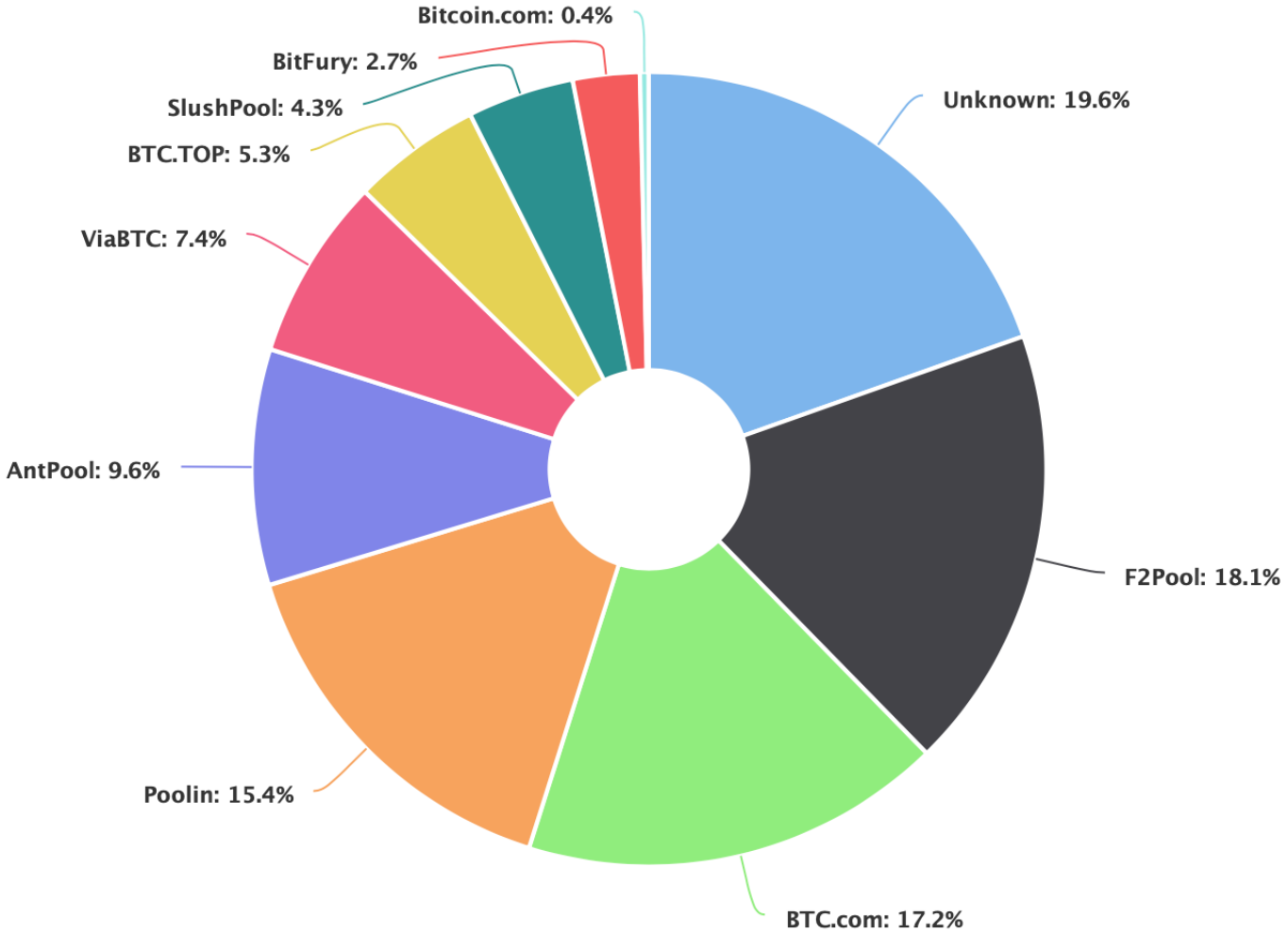


How it works?

Ex: 51% Attack

- A miner or a group of miners controls 51% or more of the mining power (hash power) of the blockchain network
- Once a group has majority control over transactions on a blockchain network, it can prevent specific transaction or even reverse older transactions -> They can double-spent!!

Is Bitcoin Decentralized?



3 Mining pools can collectively attack the system

Double Spending Attacks

How it works?

Ex: Race Attack - **reversing existing transactions by broadcasting a new chain**

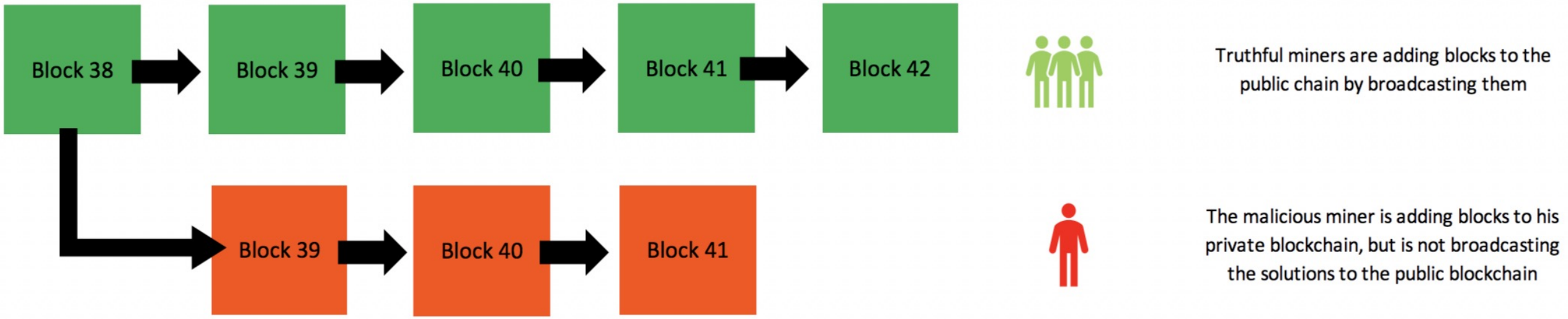
- If a merchant don’t wait for confirmations of payment, there’s a 50% chance adversary got the double spent coin
- The customer can trick the merchant if he/she sends the same coins again to his/her address.

Double Spending Attacks - How to mitigate?

- In reality these attacks are extremely hard to perform because of the requirement of "hash power" for Proof of Work
  - Adversary needs to put way too much effort for what it will give the attacker in return

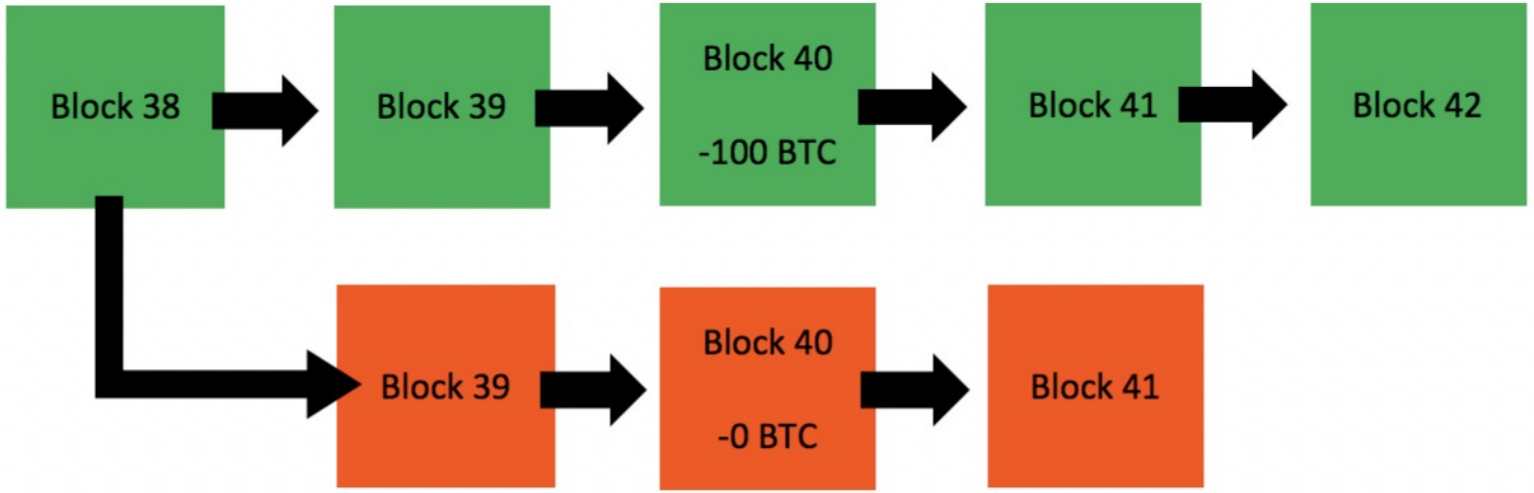
- It is recommended for merchants to wait for a **minimum of 6 confirmations**
  - Here, "6 confirmations" simply means that after a transaction was added to the blockchain, 6 more blocks should be added after it.

Double spending illustration



https://medium.com/coinmonks/what-is-a-51-attack-or-double-spend-attack-aa108db63474

Double spending illustration

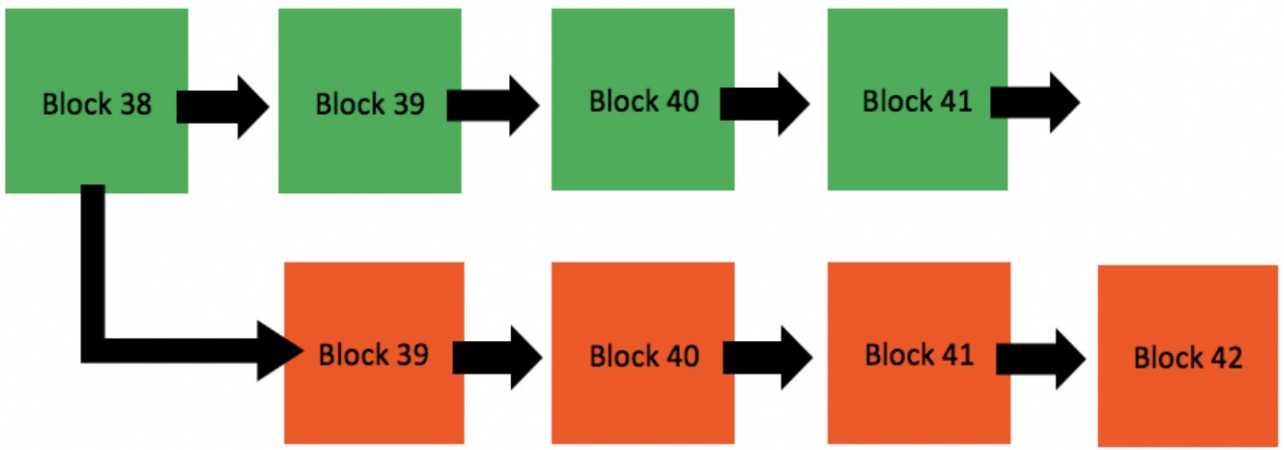


The malicious miner spends his Bitcoins on the truthful public chain on a luxurious car

Meanwhile, the malicious miner does not add this transaction to his private blockchain, on this blockchain he still possesses those Bitcins

https://medium.com/coinmonks/what-is-a-51-attack-or-double-spend-attack-aa108db63474

Double spending illustration



Truthful miners are adding blocks to the public chain, but in a considerably slower pace than the malicious miner is adding blocks to his private and stealth blockchain

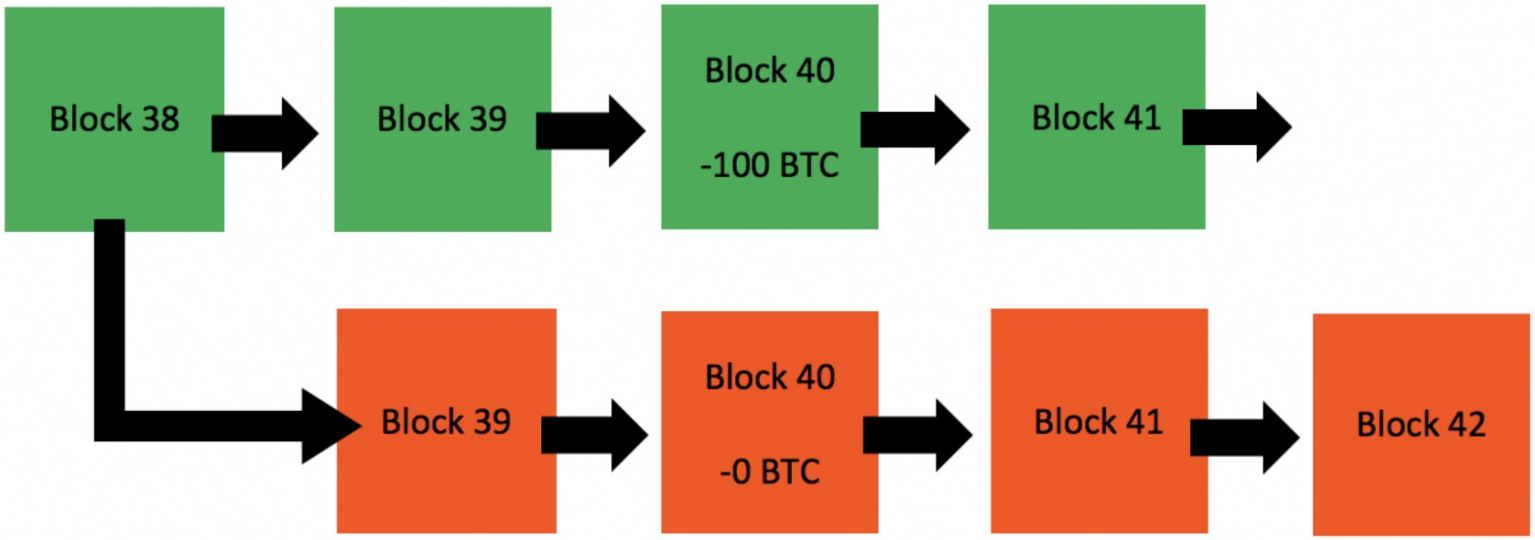
Hashing power

The malicious miner is adding blocks to his private blockchain faster, trying to catch up with the private blockchain

Hashing power

https://medium.com/coinmonks/what-is-a-51-attack-or-double-spend-attack-aa108db63474

Double spending illustration

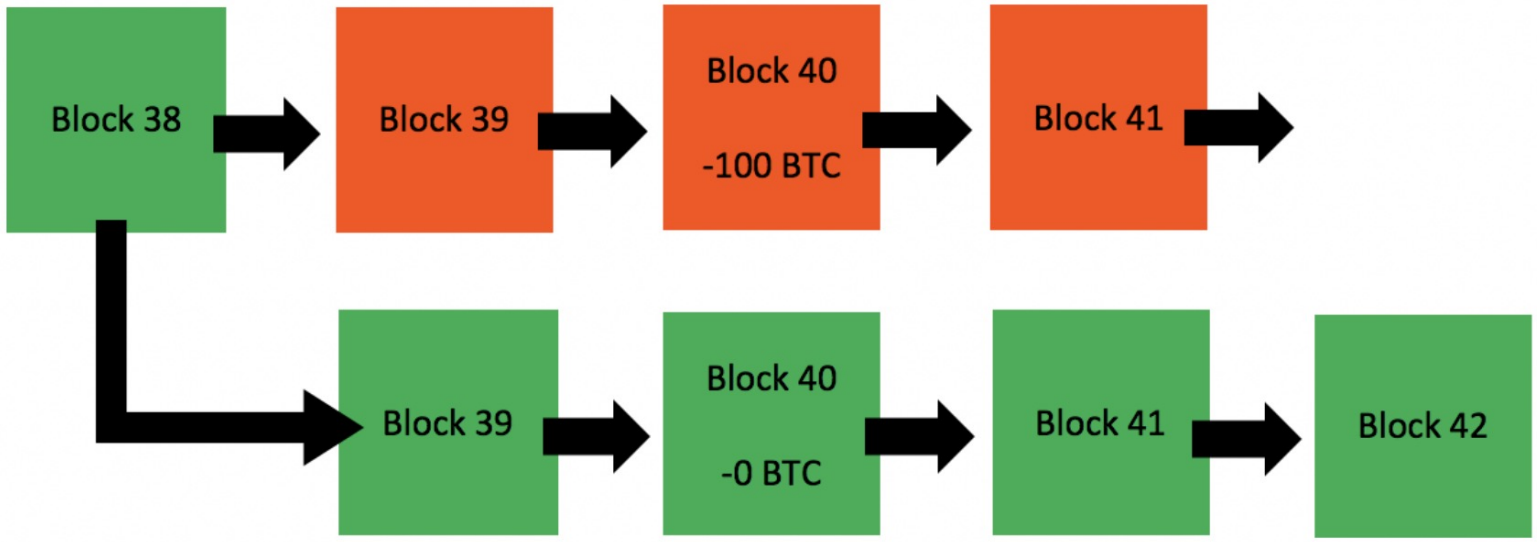


Truthful miners always follow the longest version of the chain because of the blockchain governance model, and thus they will join the malicious miner on his chain

The malicious miner broadcasts his longer version of the chain to the other miners, all wallet balances and previous transactions are now updated according to his chain because it is the longest chain

https://medium.com/coinmonks/what-is-a-51-attack-or-double-spend-attack-aa108db63474

Double spending illustration



The old public chain is abandoned because it is shorter, its data is now irrelevant

The malicious miner is once again in control of his Bitcoin, being able to spend them *again*

https://medium.com/coinmonks/what-is-a-51-attack-or-double-spend-attack-aa108db63474

The effects of the longest chain rule

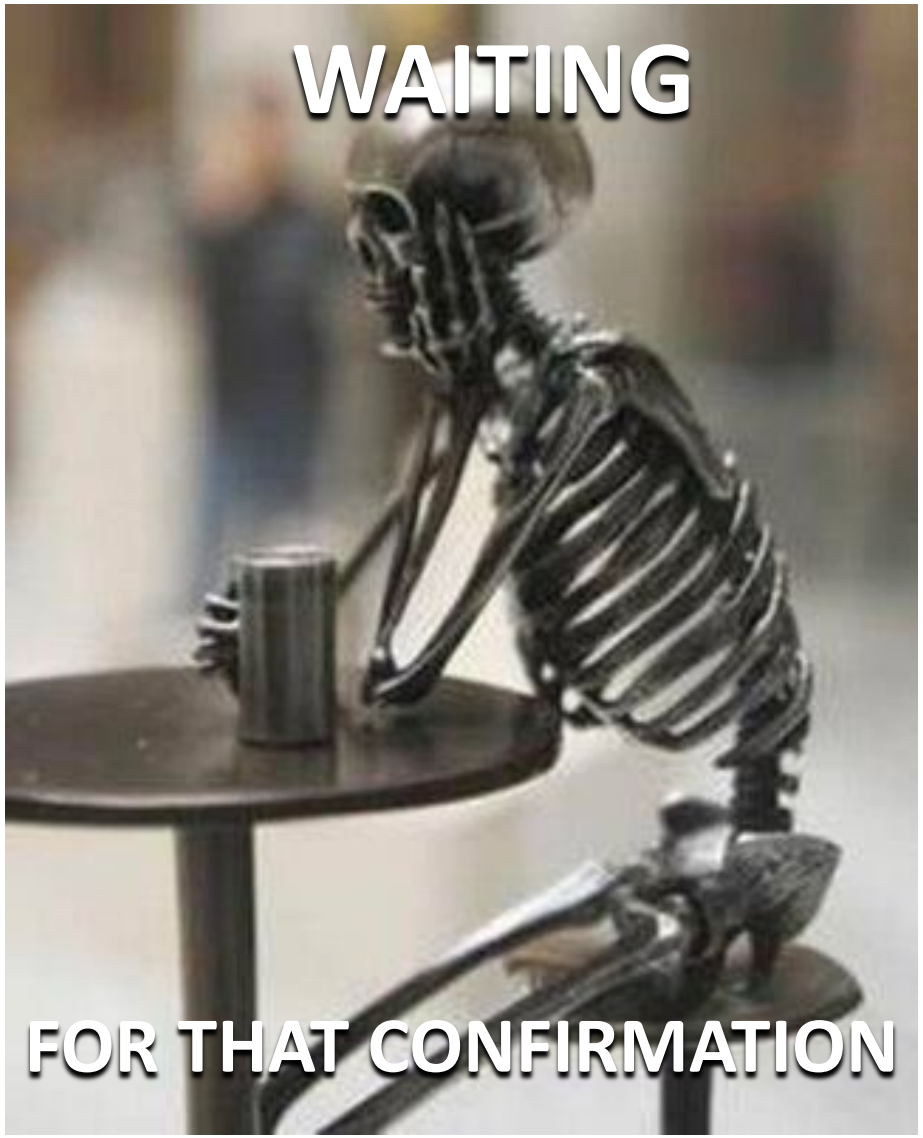
- Consensus with high probability
  - Creating blocks is hard
- The number of miners does not affect the results
- Transactions can be revoked

Risk or Wait

In order for a transaction to be valid it needs to be confirmed by the blocks.

- Each confirmation takes **10 minutes**
- Wait **one hour** to spend your money
- Real time transactions are risky, **double-spending** them is not a hard thing to do.

Be a smart merchant! Wait for the confirmation!



It is suggested that merchants should wait for at least 6 **confirmations** meaning “after the addition of transaction to the blockchain, 6 more blocks should be added”

Although you may be waiting for nothing:

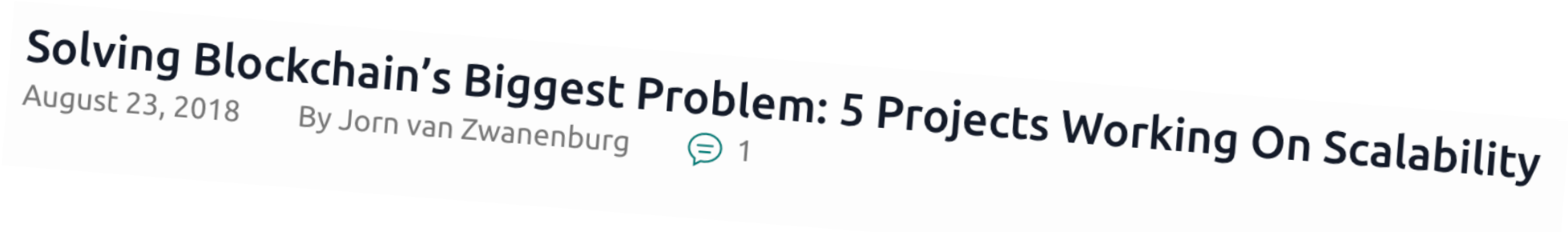
- transaction might fail in getting confirmations, if already added to another block!

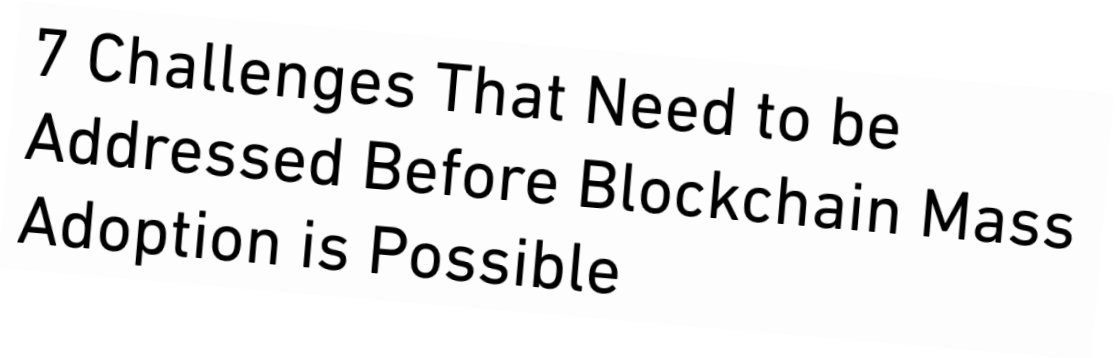
Double-spending or double-benefit? ☺

- Because of consensus algorithm, double spending is theoretically possible but highly improbable
- It may be possible however, to take advantage of a merchant providing the service or goods without waiting for the confirmation.

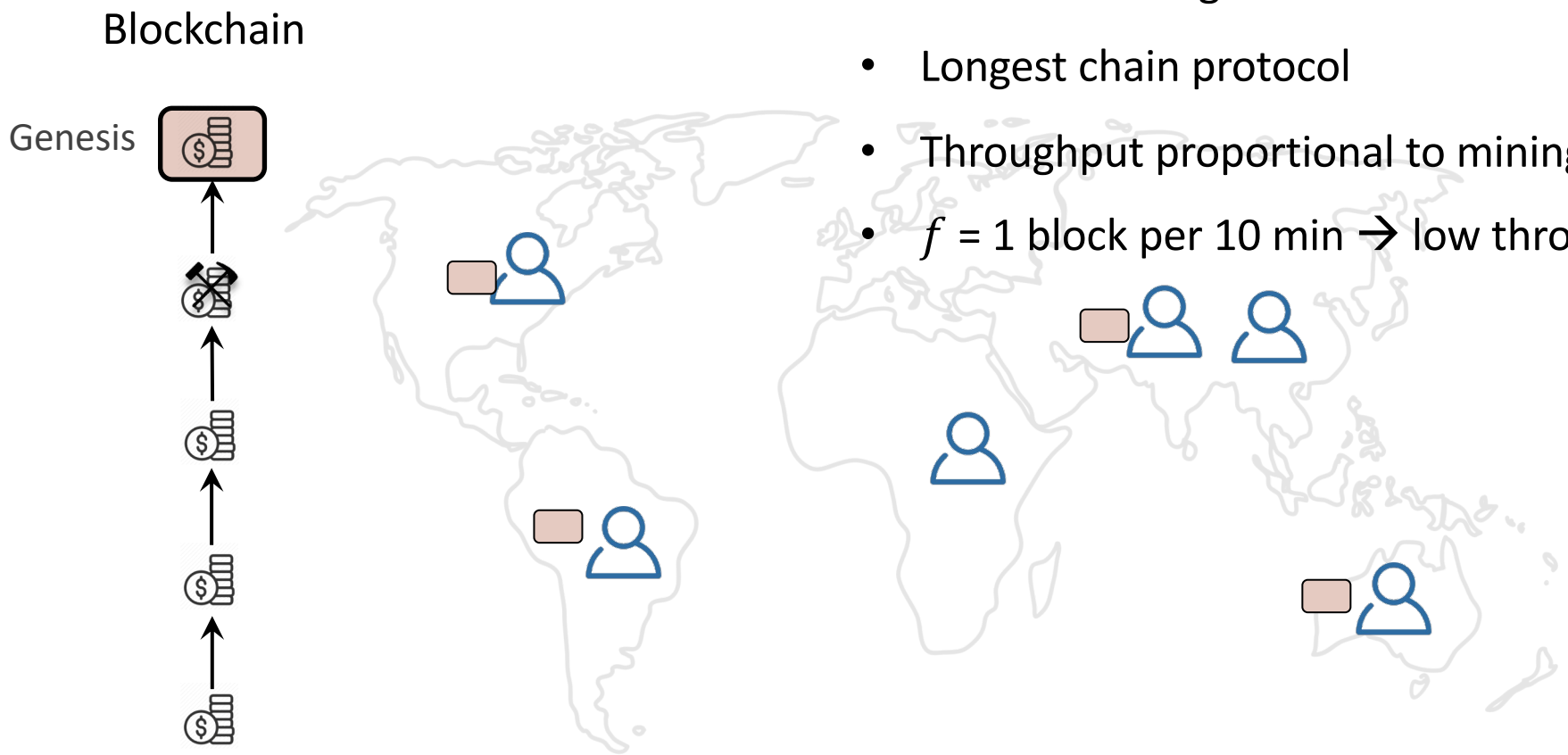
Scalability?

Principal challenge: Scalability





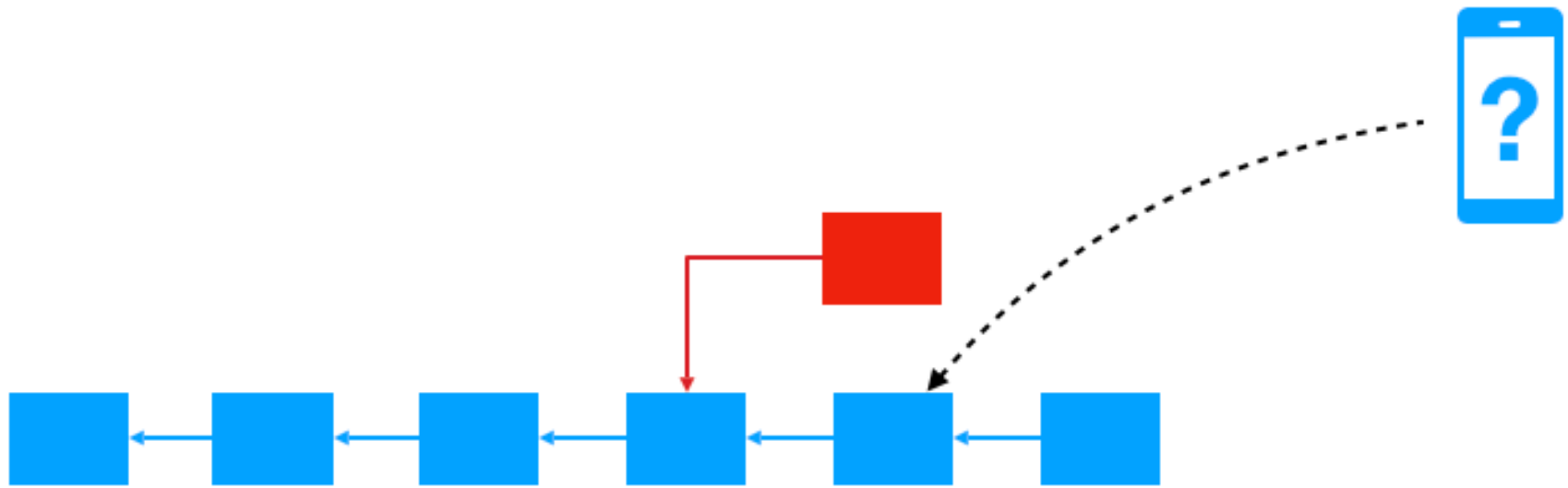
Bitcoin: A Distributed ledger



- Random mining time
- Longest chain protocol
- Throughput proportional to mining rate f
- f = 1 block per 10 min → low throughput

Problem: Efficient Verification

- How does a “light” (low-power, mobile) client securely confirm a recent (or old) transaction?
  - Especially after being offline for months, years?
  - Without “just trusting” central party (exchange)?



Alternative: Proof-of-Stake (PoS)

- **Proof-of-Stake:** assigns consensus shares in proportion to prior capital investment
  - Could address energy waste problem
  - Major unsolved security & incentive problems
- But PoS requires secure public randomness…

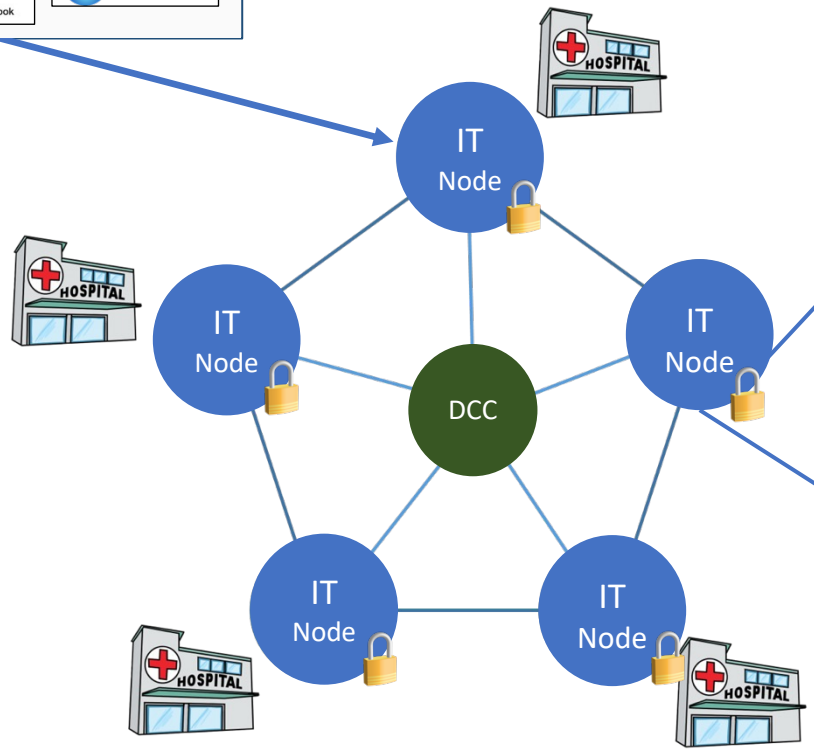
Alternative: Permissioned Ledgers

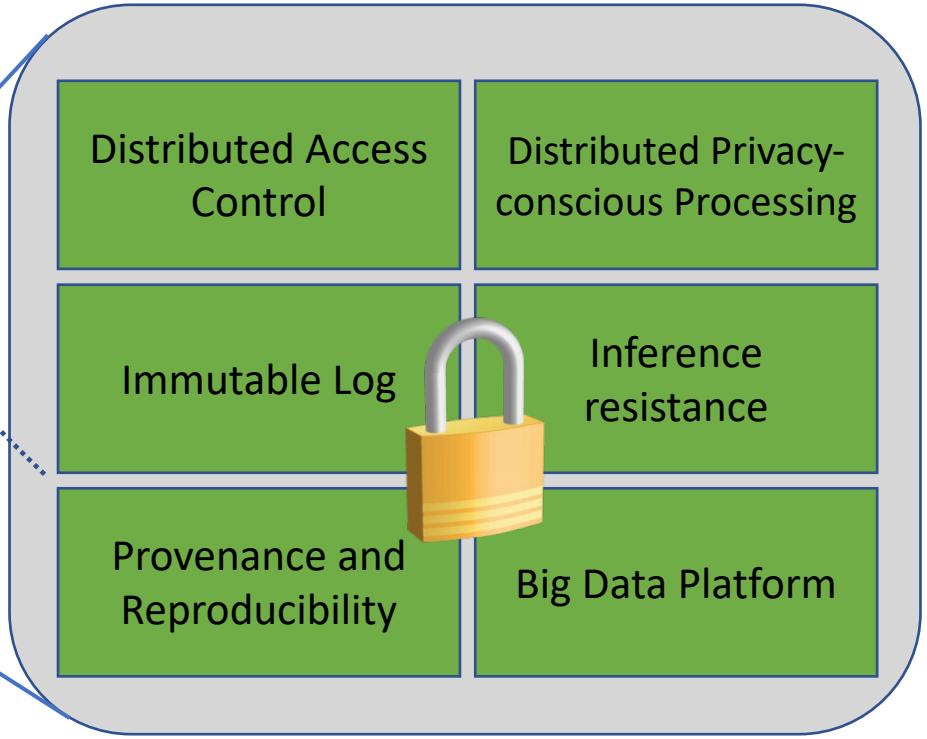
- Also called “permissioned blockchains” or “private blockchains”
- Just decide **administratively** who participates; Fixed or manually-changed group of trustees
- **Liability clearly defined**
- No proof-of-work needed → low energy cost
- More mature consensus protocols applicable
- Higher human organizational costs
- No longer open for “anyone” to participate

Tremendous potential for regulated sectors such as finance and health

Blockchain in Healthcare: Logging of Data Access

DPPH Blockchain



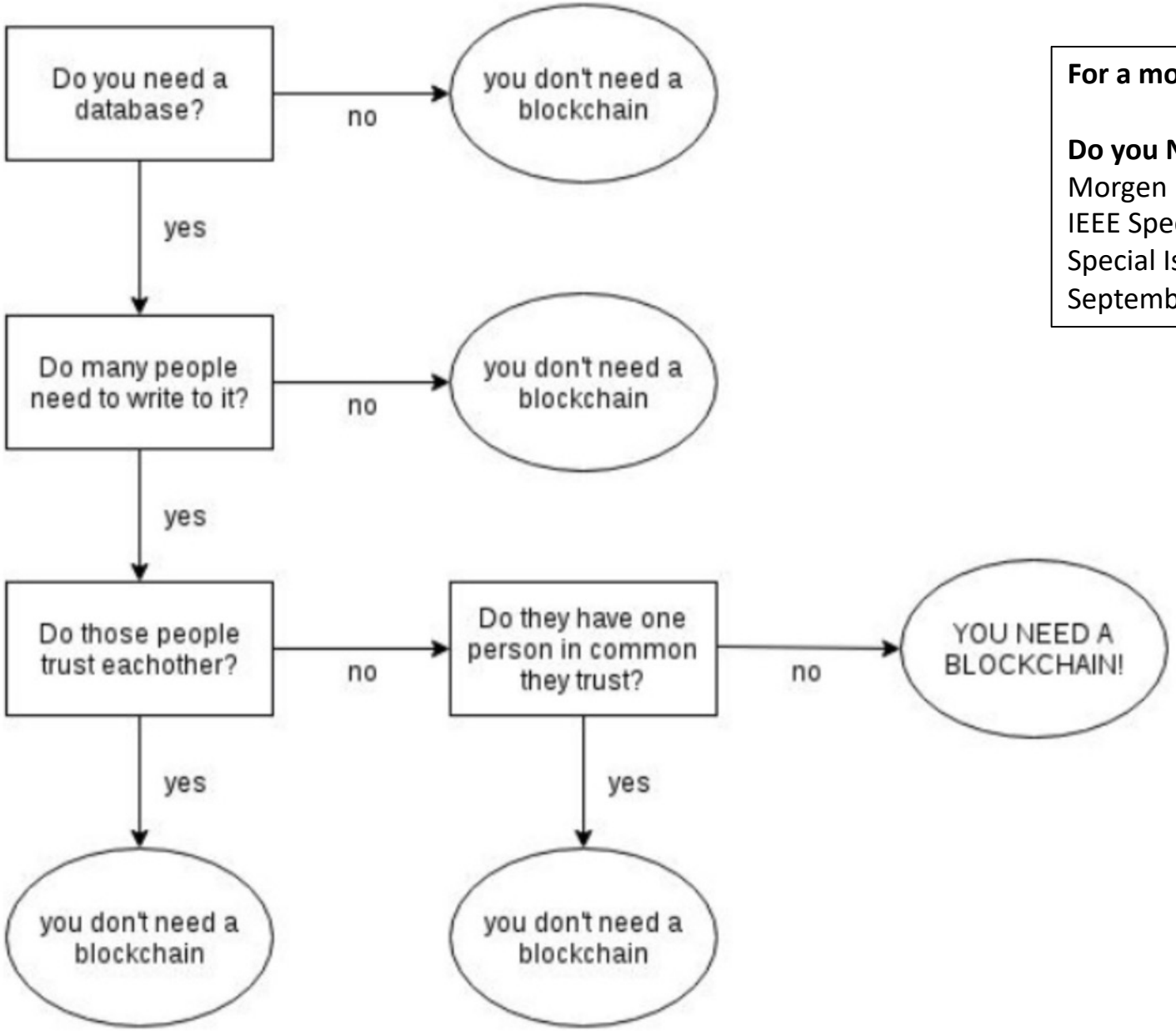


Only permissioned blockchains
Make sense in such a context

DCC: Data Coordination Center

Other application: smart contracts to support patient consent

Do You Need a Blockchain?



For a more complete version:

Do you Need a Blockchain?
Morgen E. Peck
IEEE Spectrum
Special Issue on Blockchains
September 2017

Summary

- Blocks are connected and refer to previous block with hash functions
- Miners gather transactions and bundle them to create blocks
- Blocks are added to the chain by solving a hard cryptographic puzzle -> Proof-of-Work
- Consensus is a must in blockchain to maintain the state of the blockchain, making the system fault-tolerant, and robust to attacks

Why not use blockchain in any arbitrary contracts/relations between people? -> Smart Contracts

Let's look how smart the smart contracts are!

Additional resources

Lectures by Andrew Miller:

http://soc1024.ece.illinois.edu/teaching/ece598am/fall2016/

Online Coursera Course:

https://www.coursera.org/learn/cryptocurrency

Bitcoin and cryptocurrencies book, with many videos – Narayanan, Bonneau, Felten, Miller (2016):

- http://bitcoinbook.cs.princeton.edu

Anonymity on Bitcoin

Anonymity on Bitcoin

- Many people using Bitcoin don't want others to know who they are
- Connect to Bitcoin network via Tor – is this enough?
- To obfuscate the link between individual and transaction, a different bitcoin address for each transaction can be used – is it user friendly?
  - One person could hold multiple addresses in his wallet, and in theory, there would be nothing to link those addresses together
- External laundry (mixing) services
  - Allow users to trade bitcoins whose transaction history implicates them for coins with different transaction histories
  - E.g. BitLaundry

BitLaundry

BitLaundry - For all your Bitcoin washing needs.

How BitLaundry works

BitLaundry is designed to help unlink accounts from each other. It does that by providing a well-known, and hopefully popular service. Here's how it works:

1. Imagine that Alice wishes to send Bitcoins to Bob.
2. Bob, sadly, is not well liked. Alice would rather not have anyone know that she sent Bob Bitcoins.
3. So, Alice enters Bob's Bitcoin address into the form at BitLaundry, and selects a delivery schedule.
4. Alice gets a one-time-use address from BitLaundry.
5. Alice sends her Bitcoins to that address, and they get all mixed up with BitLaundry's other Bitcoins.
6. BitLaundry waits until Alice's Bitcoins are received with 10 confirmations.
7. BitLaundry deletes the database link between the one-time-use address and Bob's address.
8. BitLaundry sends Bitcoins out to Bob according to the delivery schedule.

How it works

Tips

Send Bitcoins to yourself to obscure their history.

Use multiple recipient addresses and/or spread the transactions over a number of days to thwart correlation attacks!

Fees

2.4900% of the total you send, plus **BTC 0.00149** per outgoing transaction

Example: Let's distribute BTC 10 to 7 recipients over 3 days with 2 transactions per recipient per day:

The base fee:
BTC 10 * 0.0249 = BTC 0.2490

plus the transaction fee:

(Not So Much) Anonymity on Bitcoin

- Everything that happens in the Bitcoin world is trackable
- Every Bitcoin-based transaction is logged in the blockchain
  - If you choose to engage in sensitive transactions on Bitcoin, you should be aware that a record will be preserved for all eternity
- If your Bitcoin address goes public, everyone in the world will know your bitcoin balance and transactions
  - Can also be inferred using auxiliary information about a specific target (time of transaction, merchant, etc.)
  - Inferring personal information by analyzing large datasets is not far fetched (remember Netflix or MA Governor)
- Privacy is not enforced by the Bitcoin protocol design

What happens in the blockchain stays in the blockchain

Why Existing Methods are not Good Enough?

- Create a lot of Bitcoin addressed
  - Who has time for that?
  - Correlation between the addresses can be inferred
- Use an external laundering service
  - Laundry must be trusted
  - The service can be malicious (no anonymity)
  - The service can go out of business
  - Bitcoins can be stolen
- Users usually don’t want to expand continual effort in protecting their privacy
- Users are typically not sufficiently aware of their compromised privacy

Evaluating User Privacy in Bitcoin

Androulaki et al. 2013

- **Goal:** Evaluate the privacy that is provided by Bitcoin
- Investigates the behavior of Bitcoin client and exploiting its properties
- Through a novel simulator that mimics the use of Bitcoin as the primary currency within a university setting

Adversary Model

- Adversary *A* does not only have access to public log, but is also part of the Bitcoin system
- Can also incur one or more transactions through Bitcoin
- Can have access to the (public) addresses of some vendors along with (statistical) information
  - Such as the pricing of items or the number of their clients within a specified amount of time
- Computationally bounded

Quantifying Privacy in Bitcoin

- *Activity unlinkability*
  - An adversary *A* should not be able to link two different addresses (address unlinkability) or transactions (transaction unlinkability)
- *Profile indistinguishability*
  - (in-)ability of *A* to reconstruct the profiles of *all the users* that participate in pubLog
- Defined both definitions as a game between the adversary and a challenger
- Quantified user privacy in terms of adversary's advantage in winning the games

Tools - Heuristics

- Multi-input transactions
  - When the BTC amount is not enough in one address, multiple addresses (in one's wallet) are used for a transaction
  - If these BTCs are owned by different addresses, then the input addresses belong to the same user
- Shadow addresses
  - Automatically created and used to collect back the "change" that results from any transaction issued by the user
  - When a Bitcoin transaction has two output addresses *x* and *y*, and if *x* has appeared in the public log before:
    - *y* is a shadow address

Tools – Behavioral Analysis

- *Adversary A* also uses behavior-based clustering techniques
  - K-Means (KMC), Hierarchical Agglomerative Clustering (HAC) algorithms
- Goal: Output a group of clusters of addresses that approximates the Bitcoin users the best
  - Utilizing the similarities between Bitcoin addresses
  - Time of transaction, value, etc.
- HAC is applied on top of the results received using heuristics
- KMC is applied on top of results obtained via HAC
- Refer to the paper for details

Results and Limitations

- Behavior-based clustering techniques can unveil the profiles of 40% of Bitcoin users
  - With 80% accuracy
  - Even if these users try to enhance their privacy by manually creating new addresses
- Limitations:
  - Experimental setup (not real data)
  - Laundry service is not considered as a privacy tool

Zerocoin and Zerocash

Zerocoin: Towards Anonymous Bitcoin

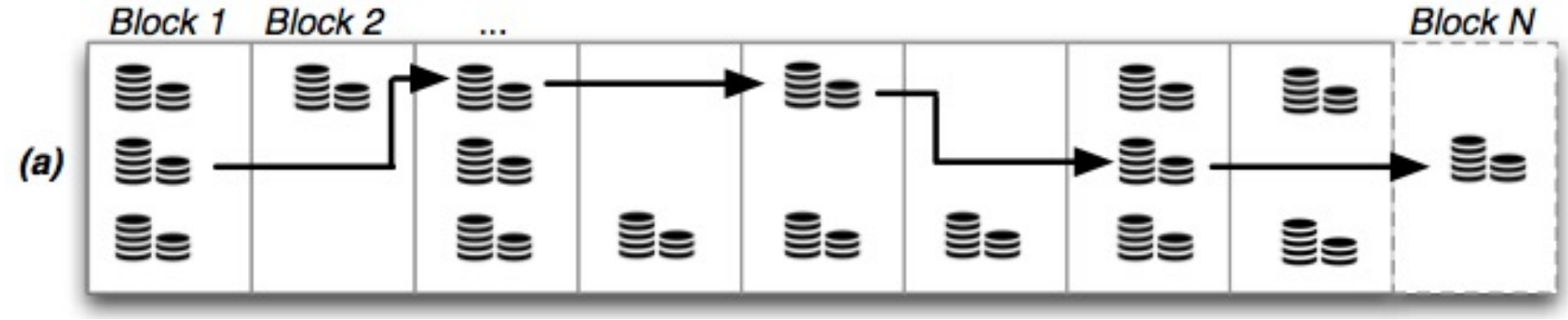
Miers et al. 2013

- Effectively builds a money-laundering service into a crypto currency at the *protocol level*
  - Eliminates any reliance on trusted third parties
- Zerocoins are purchased with Bitcoin in fixed denominations
  - *Zerocoin mint*
- System takes original Bitcoins, turns them into Zerocoins, and then turns them back into new Bitcoins in another wallet
- Anyone with a Bitcoin can spend it to create a Zerocoin
  - There is a serial number inside of every Zerocoin
  - Each Zerocoin is like the encryption of that serial number
- Users can come back at any time to redeem their Bitcoins

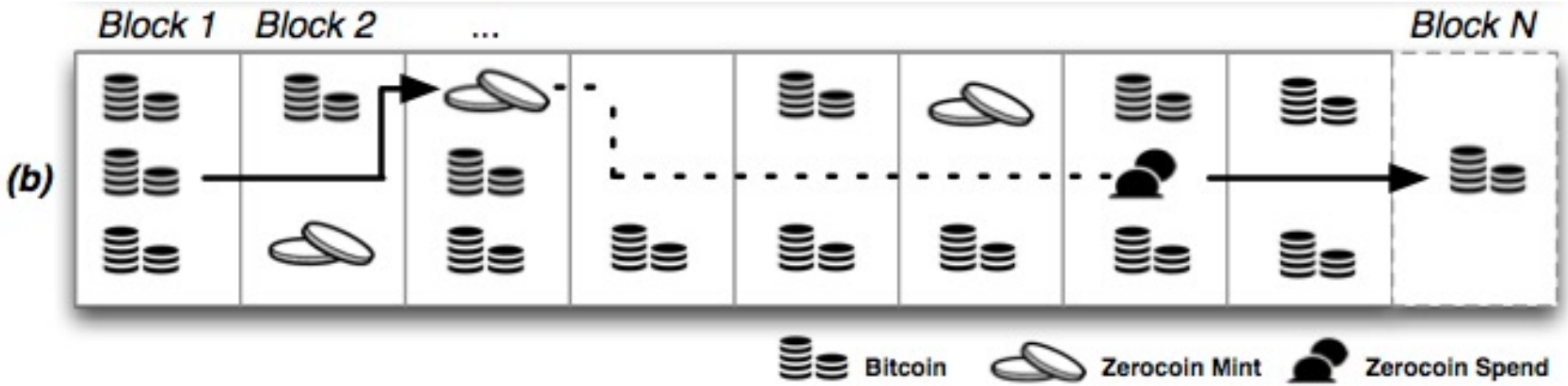
Zerocoin - Simplified (by Matthew Green)

- People throw dollars into a hat
- Each time they throw a dollar, they get a token
  - All the tokens look exactly the same
- Bob comes back with a mask on
  - Or gives his token to a friend and he goes back
- Bob exchanges his token, and he takes out a totally different dollar

Zerocoin vs. Bitcoin



Bitcoin block chain. Each transaction is tied to the one that precedes it



Bitcoin/Zerocoin block chain. A user transforms Bitcoins into a Zerocoin, then “Spends” it to redeem the Bitcoins. The linkage between Mint and Spend (dotted line) cannot be determined from the blockchain data

Figure: Miers et al. Zerocoin: Anonymous Distributed E-Cash from Bitcoin.

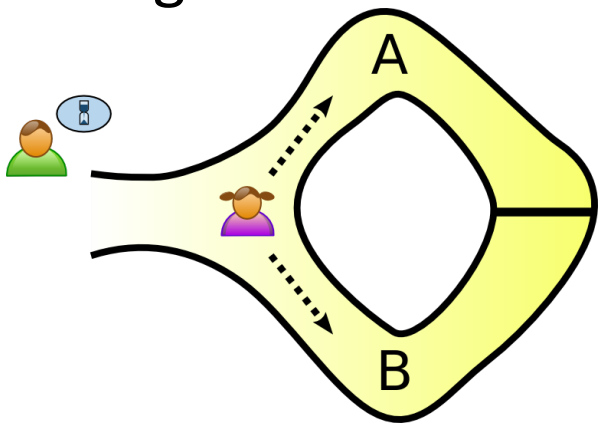
Zerocoin - Challenge

- How do people get their coins back without leaving their fingerprints all over it

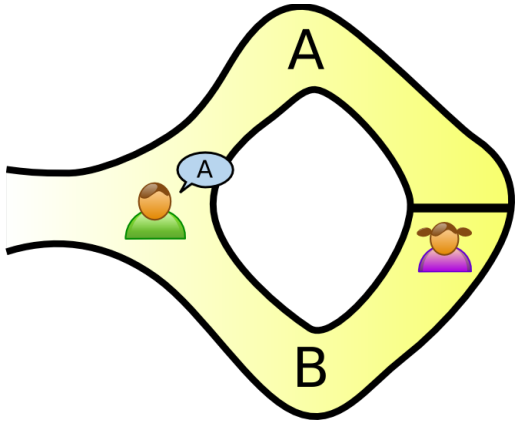
- **Solution:** zero-knowledge proof
  - A party can prove to another that a given statement is true, without conveying any additional information apart from the fact that the statement is indeed true
- Zero-knowledge proof says two things:
  - I am an owner of a Zerocoin and I know a serial number that is inside of the coin I made
  - I actually paid for the Zerocoin

Reminder: Zero-Knowledge Proofs

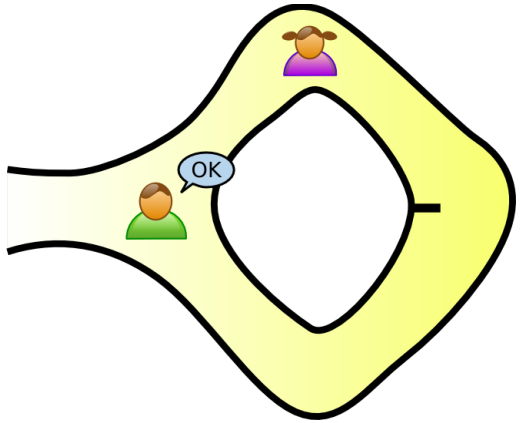
- Peggy has uncovered the secret word used to open a magic door in a cave
- Victor wants to know whether Peggy knows the secret word; but Peggy, being a very private person, does not want to reveal the fact of her knowledge to the world in general



Peggy randomly takes either path A or B, while Victor waits outside



Victor chooses an exit path



Peggy reliably appears at the exit Victor names

Figure: Wikipedia

Zerocoin - Protocol

- Goal: build a crypto currency where your neighbors, friends and enemies can’t see what you bought or for how much
- **Zerocoin *mint* and *spend* transactions**
- Routine day-to-day transactions must be conducted via Bitcoin
  - Due to performance and functionality drawbacks

Minting a Zerocoin

- Bob generates a random serial number S, and encrypts this into a coin C by use of second random number r
- Coin C is added to a cryptographic accumulator by miners
- The amount of the base currency (Bitcoin) equal in value to the denomination of the Zerocoin is added to a Zerocoin escrow pool

Redeeming (Spending) a Zerocoin

- Bob needs to prove two things via a zero-knowledge proof
  - He knows a coin C that belongs to the set of all other minted Zerocoins $(C1, C2, \ldots\ Cn)$, without revealing which coin it is
    - By use of a one-way accumulator that does not reveal the members of the set
  - He knows a number r, that along with the serial number S corresponds to a Zerocoin
- The proof and serial number S are posted as a Zerocoin spend transaction
- Miners verify the proof and that the serial number S has not been spent previously
- If verified, the transaction is posted to the blockchain
- The amount of the base currency equal to the Zerocoin denomination is transferred from the Zerocoin escrow pool

Zerocoin - Anonymity

- Anonymity in the transaction is assured because the minted coin C is not linked to the serial number S used to redeem the coin
  - Serial number S is only publicly revealed after a redeem operation
- Limitations:
  - Size of the anonymity set depends on the number of coins minted by honest users
  - Reveals number of minted and spent coins to all users

Zerocoin - Drawbacks

- Efficiency Issues:
  - Extra computation time required by the process (to be performed by the miners)
    - The verification time for a block increases
    - Authors show that that verification time for an entire block would not exceed five minutes
  - If the proofs were posted to the blockchain, the size of the blockchain dramatically increases
    - Authors stated the proofs can be stored outside of the blockchain
- Not adapted by Bitcoin
  - Above efficiency issues
  - Political reasons…

Zerocoin - Drawbacks

- Anonymity problems:
  - Reveals payments' destinations and amounts
- Functionality problems:
  - Does not support payments of exact values
  - Does not provide a functionality to divide coins
- Does not support direct transfer of Zerocoins between users

Zerocash

Ben-Sasson et al. 2014

- Functions on top of any ledger-based base currency, such as Bitcoin
- More efficient than Zerocoin
  - Zerocash transactions are less than 1KB and take less than 6ms to verify
  - Zerocoin transactions exceeds 45 kB and require 450 ms to verify
- Hides the amount of the payment and destination
- All transactions can be made in terms of Zerocoins
- Users can:
  - Convert from Bitcoins to Zerocoins
  - Send Zerocoins to other users
  - Split or merge Zerocoins they own

Zerocash – Protocol

- *Mint transactions* and *pour transactions*
- Zerocash does not use any zero-knowledge proof
- Leverages ***zero-knowledge Succinct Non-interactive ARguments of Knowledge*** (zk-SNARK) systems
  - Zero-knowledge proofs that are particularly short and easy to verify

Discussion– Too Much Anonymity?

- **Concern:** decentralized anonymous payments will facilitate laundering of ill-gotten funds by criminals
  - Anonymity offered by Zerocash may facilitate illegal activity
- Arguments against the concern (by the authors):
  - Main difficulty with money laundering does not lie in how to privately transfer money from one person to another, but in how to make the eventual income appear legitimate
  - Even without the “help” of Zerocash, criminal users can already anonymize their activities via existing financial systems (e.g., by using cash)
- **A simple solution:** A backdoor could be added to allow police to track money laundering

References

- S. Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System, 2008
- E. Androulaki, G. Karame, M. Roeschlin, T. Scherer, and S. Capkun. Evaluating User Privacy in Bitcoin. In Proceedings of Financial Cryptography 2013.
- I. Miers, C. Garman, M. Green, and A. D. Rubin. Zerocoin: Anonymous Distributed E-Cash from Bitcoin. In Proceedings of the IEEE Symposium on Security and Privacy, 2013.
- Ben-Sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., Virza, M. Zerocash: Decentralized anonymous payments from Bitcoin. In Proc. of the 35th Symposium on Security and Privacy. S&P'14.