BLOCKCHAIN: SMART CONTRACTS LECTURE 1 - INTRODUCTION

FLORIN CRACIUN

IMPORTANT

Some of the following slides are the property of

Dr. Emanuel Onica & Dr. Andrei Arusoaie

Faculty of Computer Science,

Alexandru Ioan Cuza University of Iași

and are used with their consent.

CONTENTS

COURSE OVERVIEW

The course will cover multiple blockchain related topics.

... but not how to get rich from crypto trading

Particular focus in labs: Ethereum platform and smart contracts

Curriculum overview:

- **Blockchain basics**
- **Ethereum and smart contracts**
- **Solidity**
- **Contract patterns and verification**
- **DApps examples**
- **Bitcoin network and applications**
- **Consensus in blockchain**
- **Advanced architecture details: mining, storage and communication**
- **Security in blockchain**
- **Hyperledger Fabric and other platforms**

COURSE OVERVIEW

Bibliography

Books:

- ***Mastering Ethereum: Building Smart Contracts and DApps* – A.M. Antonopoulos, G. Wood – O'Reilly Media, 2018**
- ***Mastering Bitcoin* – A.M. Antonopoulos – O'Reilly Media, 2017**
- ***Blockchain Applications: A Hands-On Approach* – A. Bahga, V. Madisetti – VPT Publishing House, 2017**

Other sources (online docs, conference articles, websites):

- ***Solidity:* https://solidity.readthedocs.io/en/v0.7.0/**
- ***HLF:* https://hyperledger-fabric.readthedocs.io/en/release-2.2/**
- ***ACM Symposium on Principles of Distributed Computing (PODC)***
- ***ACM Conference on Computer and Communications Security (CCS)***
- ***https://www.coindesk.com***

COURSE RULES

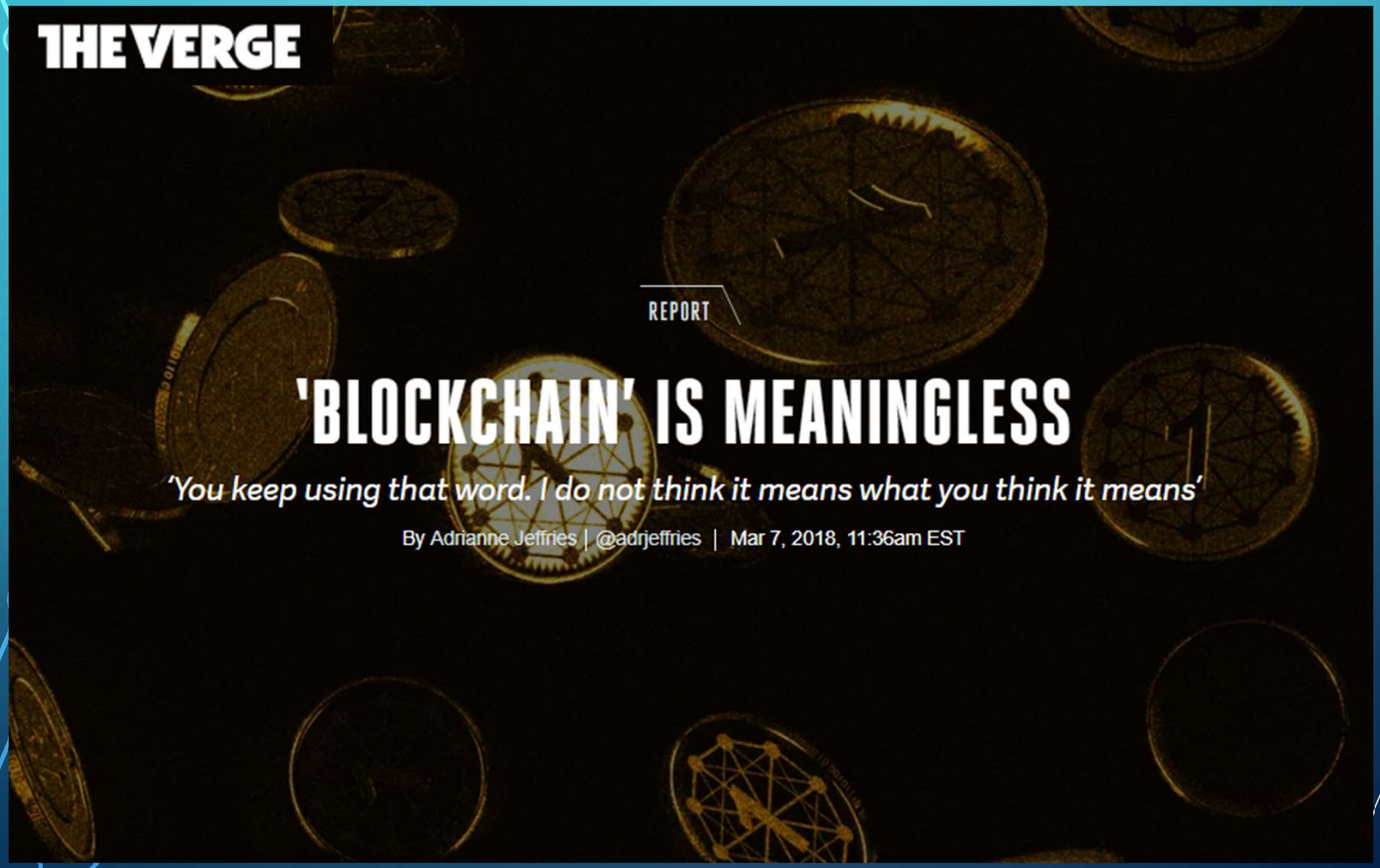
Evaluation

- Lab activity: 50% of the final grade (as it is explained at the lab)
- Final exam: 50% of the final grade (written exam, open book)
- In order to pass this course, you must get minimal 5 at the Lab and minimum 5 at the final exam
- In order to get into the final exam (including "restanta") you must follow the faculty rules regarding the lab attendance

Course team: Florin Craciun and Radu Ometita

Lab team: Radu Ometita

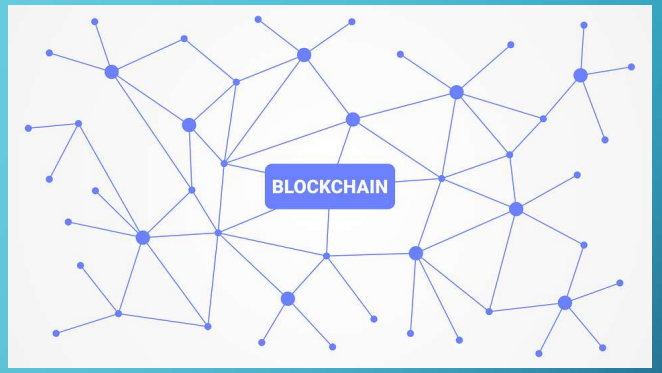
WHAT IS BLOCKCHAIN?



WHAT IS BLOCKCHAIN?

There is no accepted universal definition.

Essentially a distributed architecture capable of data storage formed of:

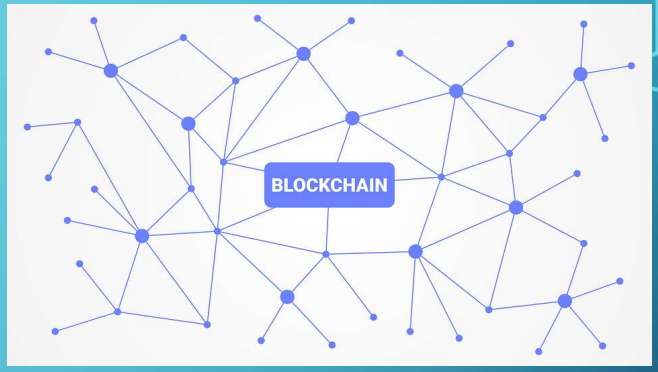


- *A ledger (book of records):* The data stored in the blockchain. A record represents a *transaction* initiated by a network participant.
- *Blocks:* Data units forming the ledger that can group multiple transactions.
- *Chaining method:* Chronologically and securely linking of the blocks into a *blockchain*.
- *Nodes (peers):* Participants in the network, typically each storing a blockchain copy (or being able to access it).

WHAT IS BLOCKCHAIN?

There is no accepted universal definition.

The architecture normally provides some specific guarantees:



- *Decentralized trust:* Network participants work together for validating transactions, without a central authority, and the platform tolerates individual failures.
- *Transparency:* Transactions are visible and trackable in the blockchain.
- *Immutability:* Once a transaction has been validated and appended in the blockchain, it cannot be changed or deleted.

WHAT IS BLOCKCHAIN?

How did this start?

- *„Bitcoin: A Peer-to-Peer Electronic Cash System"* posted on October 31st, 2008 on a cryptography mailing list
- Target: a cryptocurrency for conducting financial transactions with no trusted third party
- Author: *Satoshi Nakamoto* still remains unknown
- Open source released and first transaction conducted in January 2009

Bitcoin P2P e-cash paper

Satoshi Nakamoto satoshi at vistomail.com
Fri Oct 31 14:10:00 EDT 2008

- Previous message: Fw: SHA-3 lounge
- **Messages sorted by:** [date] [thread] [subject] [author]

I've been working on a new electronic cash system that's fully peer-to-peer, with no trusted third party.

The paper is available at:
http://www.bitcoin.org/bitcoin.pdf

The main properties:
Double-spending is prevented with a peer-to-peer network.
No mint or other trusted parties.
Participants can be anonymous.
New coins are made from Hashcash style proof-of-work.
The proof-of-work for new coin generation also powers the network to prevent double-spending.

Bitcoin: A Peer-to-Peer Electronic Cash System

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without the burdens of going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as honest nodes control the most CPU power on the network, they can generate the longest chain and outpace any attackers. The network itself requires minimal structure. Messages are broadcasted on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

Full paper at:
http://www.bitcoin.org/bitcoin.pdf

Satoshi Nakamoto

The Cryptography Mailing List
Unsubscribe by sending "unsubscribe cryptography" to majordomo at metzdowd.com

WHAT IS BLOCKCHAIN?

How did this start?

- Most concepts found in previous academic literature
- Linked timestamped records (Haber & Stornetta '97, Benaloh & de Mare '91)
- Consensus:
  - BFT (Lamport et al. '82), PBFT (Castro & Liskov '99)
  - PoW for Sybil tolerance (Dwork & Naor '92, Douceur '02), hashcash for PoW (Back '97)
- Main Bitcoin contribution: linking all needed background and incentivizing Proof-of-Work
- What do all these mean and why do we need them?

FIGURE 1: CHRONOLOGY OF KEY IDEAS FOUND IN BITCOIN

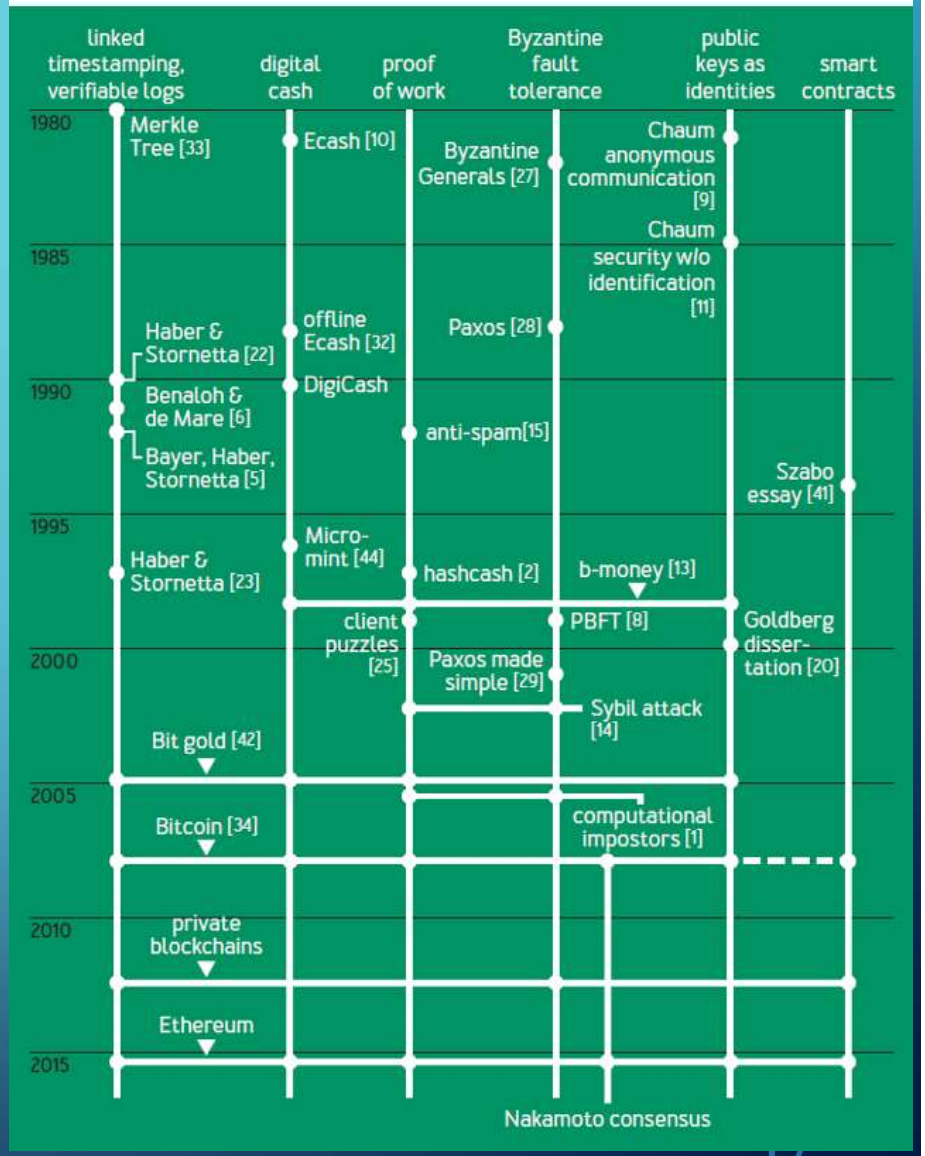
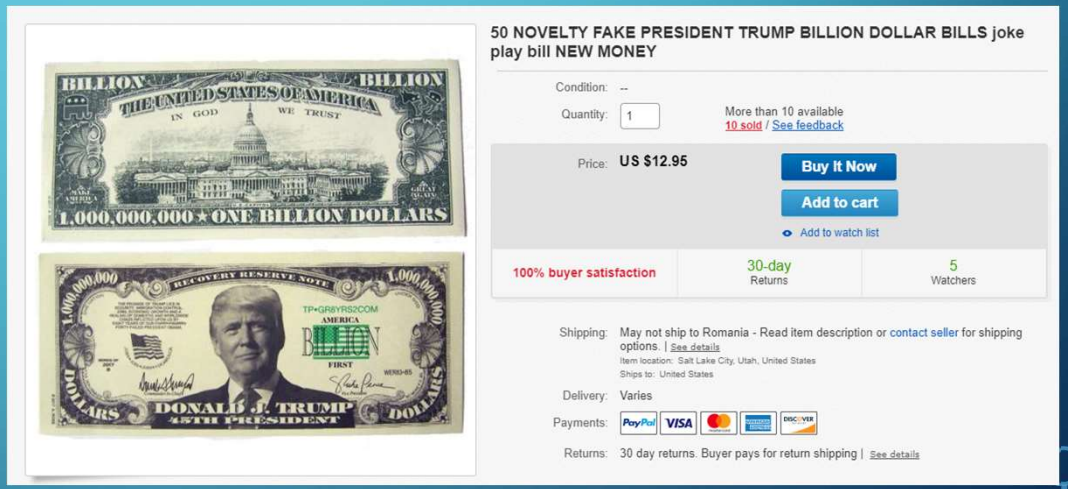


Figure source: *Bitcoin's Academic Pedigree* (A. Narayanan, J. Clark) ACM Queue, Vol. 15, Issue 4

WHAT IS BLOCKCHAIN?

- Let's stick for a while to cryptocurrencies
- A normal currency requires *trust* offered by a bank (central point). This can guarantee for:
  - Validity of money
  - Value of money
  - Transactions validity



- We'll discuss the role of blockchain in providing actual value for a cryptocurrency in a later course (maybe...)

WHAT IS BLOCKCHAIN?

- Let's stick for a while to cryptocurrencies
- Assume you buy your first „crypto money" (e.g., using some crypto exchange like bitstamp.net)
- Now you want to perform cryptocurrency transactions without the presence of a bank
- You still need a common base of trust
  - Ownership of money is guaranteed by previous transactions
  - Previous transactions must be verifiable
  - Nobody should be able to tamper with these

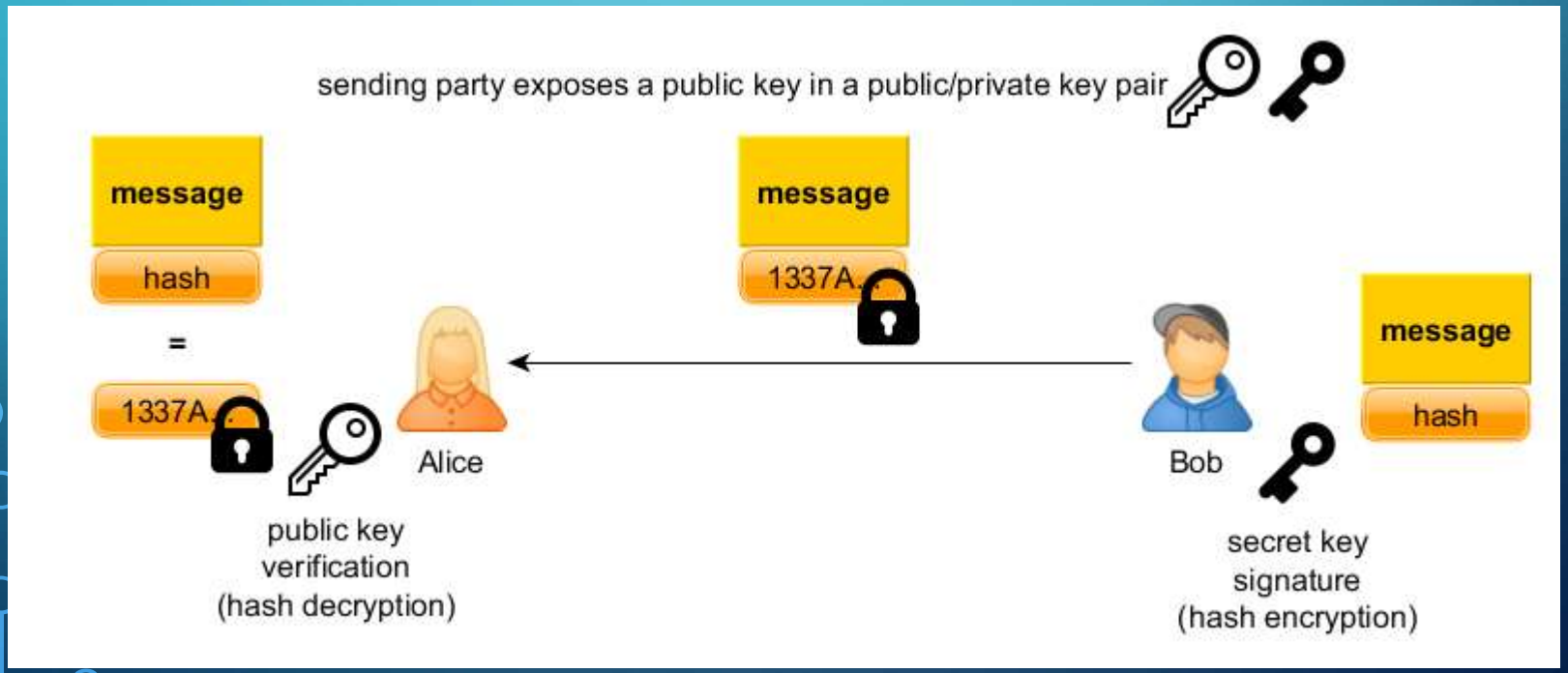
WHAT IS BLOCKCHAIN? (QUICK CRYPTO PRIMER)

- How does the blockchain architecture provide the needed trust?
- Symmetric encryption:

- Public key (asymmetric) encryption:

WHAT IS BLOCKCHAIN? (QUICK CRYPTO PRIMER)

- Digital signatures



WHAT IS BLOCKCHAIN?

- How does the blockchain architecture provide the needed trust?

- Each blockchain user has a public/private key pair
- The public key has also the role to identify the user

- **In essence,** when user A creates a transaction to pay user B:
  - A includes reference to previous transactions towards A that provided the money to be sent
  - A includes the public key of destination B
  - A signs the formed transaction with the private key and submits it to the blockchain
  - Transaction data can be verified with A's public key
  - This proves that A owns the money (transaction value)
  - This proves that the money are valid (by tracking back the source transactions)

WHAT IS BLOCKCHAIN?

- How does the blockchain architecture provide the needed trust?

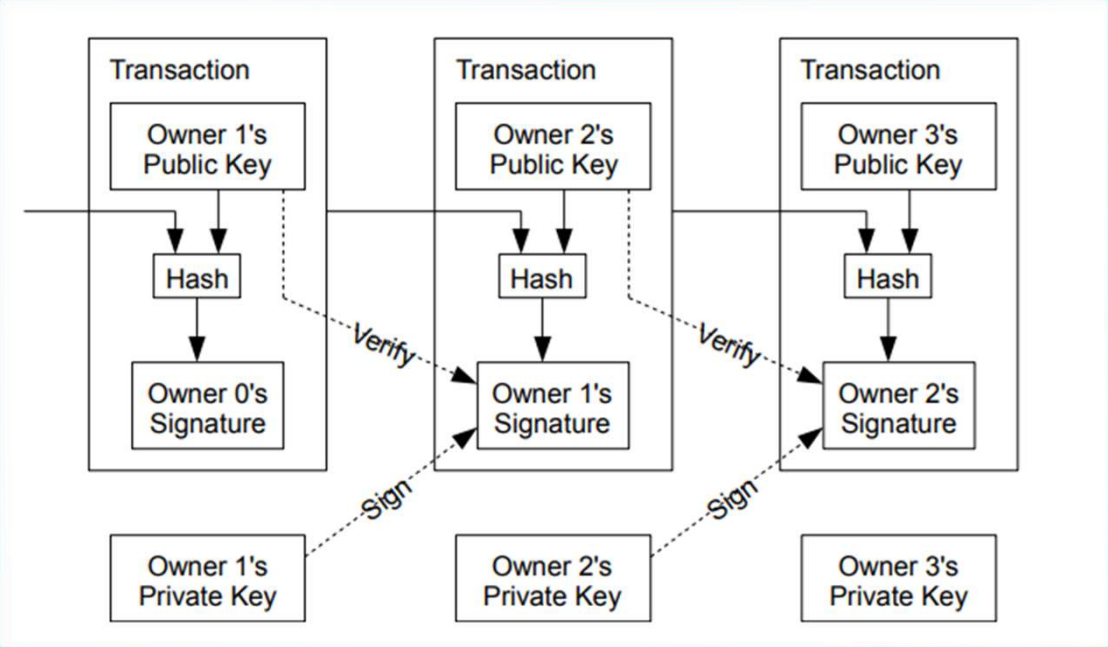


Figure source: *Bitcoin: A Peer-to-Peer Electronic Cash System* (S. Nakamoto), 2009

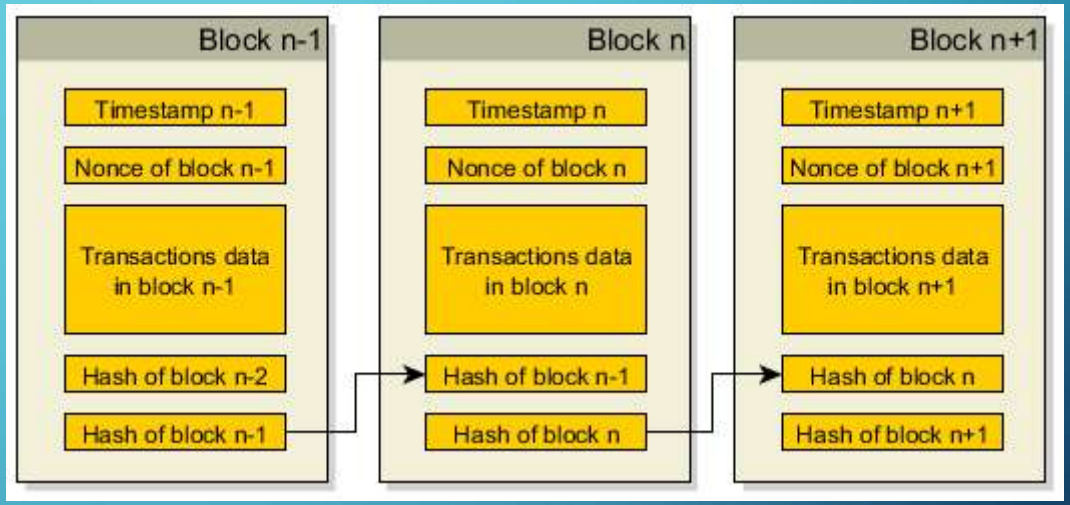
- Such linking can prove to B that A owned the money sent...
- But it cannot prove that A didn't send before the same money to C!
- The *double spending* problem: here comes the blockchain

WHAT IS BLOCKCHAIN?

- How does the blockchain architecture provide the needed trust?
- Transactions are grouped in blocks
- First key element – a cryptographic hash function:
  - maps data of arbitrary size to a fixed length digest
  - deterministic and quick to compute
  - collision resistant: hard to find two inputs with same output
  - one-way: intractable to find the input for a given output
  - examples: SHA-2 family, SHA-3 (Keccak) family, etc.
- Each block in a blockchain includes a hash digest over its contents
- Modifying any field in a block would change the hash digest
- But... This still does not prevent the double spending...

WHAT IS BLOCKCHAIN?

- Second key element – unique time ordering of transaction blocks (this is the „chain" of linked timestamp records)
- **In essence,** the structure of the blocks looks like this:



- Each block includes a timestamp/order info of the block
- Each block includes the hash of the previous block
- Timestamp/ordering proves that transactions existed at a previous time
- Each new hash in a new block re-inforces info in all previous blocks

  (changing some previous block info will result in a complete hash chain change)
- But... This still does not *completely* prevent the double spending...

WHAT IS BLOCKCHAIN?

- Blocks are added to the chain by peers, who agree on each new block
- Each peer stores (can access) a consistent blockchain copy
- Peers can verify previous transactions and reject invalid ones

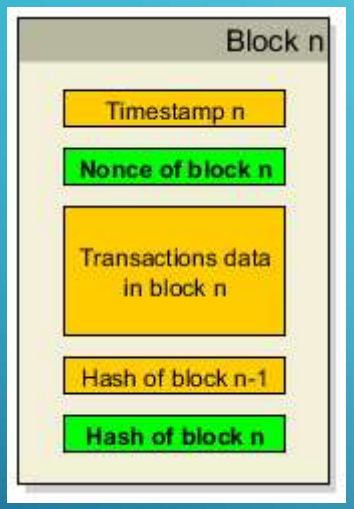
  (i.e., reject sending money to C that were previously sent to B)
- We have one single blockchain

  (i.e., no next concurrent blocks including same payment to B and C)
- But an attacker could still try to change the blockchain!
  - Alters the blockchain by modifying a transaction and all following blocks
  - Injects a majority of fake peers in the system holding the altered copy

    (Sybil attack)

WHAT IS BLOCKCHAIN?

- Third key element: proof-of-work (PoW) for confirming a new block



- Peers must *mine*: perform some hard computational puzzle to propose a new block for inclusion in blockchain
- **In essence:**
  - When a new block is proposed by a peer, the block hash is required to have the value in a particular range (e.g., starting with *k* zero digits)
  - Peers keeps trying different nonce values until the desired hash output is obtained

- Injecting fake peers power ≠ Injecting fake peers mining power
- Changing a block inflicts change in all following hash-chained blocks
- Therefore, changing a block requires re-mining all blocks until the last one
- Hard to catch up with all honest peers that work for adding a new block

WHAT IS BLOCKCHAIN?

- One final issue: why would a peer take the effort to mine a block?
- After all not all peers mine (you can use the blockchain just to perform transactions)
- Incentives: successful peers are rewarded on getting the PoW

- This theoretically, and until now also practically, motivates more honest peers (honest mining power) on getting the job done than the power an attacker could amass (should be less than 51%)

WHAT IS BLOCKCHAIN?

- The indirect benefit of proof-of-work: *consensus*

- The node finding the PoW for a block broadcasts the block to all nodes
- Nodes agree on, confirm and „chain" the new block, if (**in essence**):
  - all included transactions verify as valid (no double spending)
  - previous latest chained block hash is correct
  - current block hash value verifies the PoW condition
- Mining nodes start mining for the next block after

WHAT IS BLOCKCHAIN?

- What if two nodes find the PoW for a mined block simultaneously?

  (block fork)

- **In essence,** the longest chain wins:
  - a miner will start working on the next block based on the latest confirmed
  - if a „second latest" valid block is received this is kept on hold
  - eventually some miner will finish faster the PoW based on his „latest" block fork and broadcast the new head of chain
  - miners working over the other „latest" block fork will stop their work, discard their fork and switch to the block on hold and the new head

- Difficulty of proof-of work ensures that finding PoW simultaneously happens rarely
  - i.e., in Bitcoin one-block fork typically not more than once/week, two-blocks fork much rarely

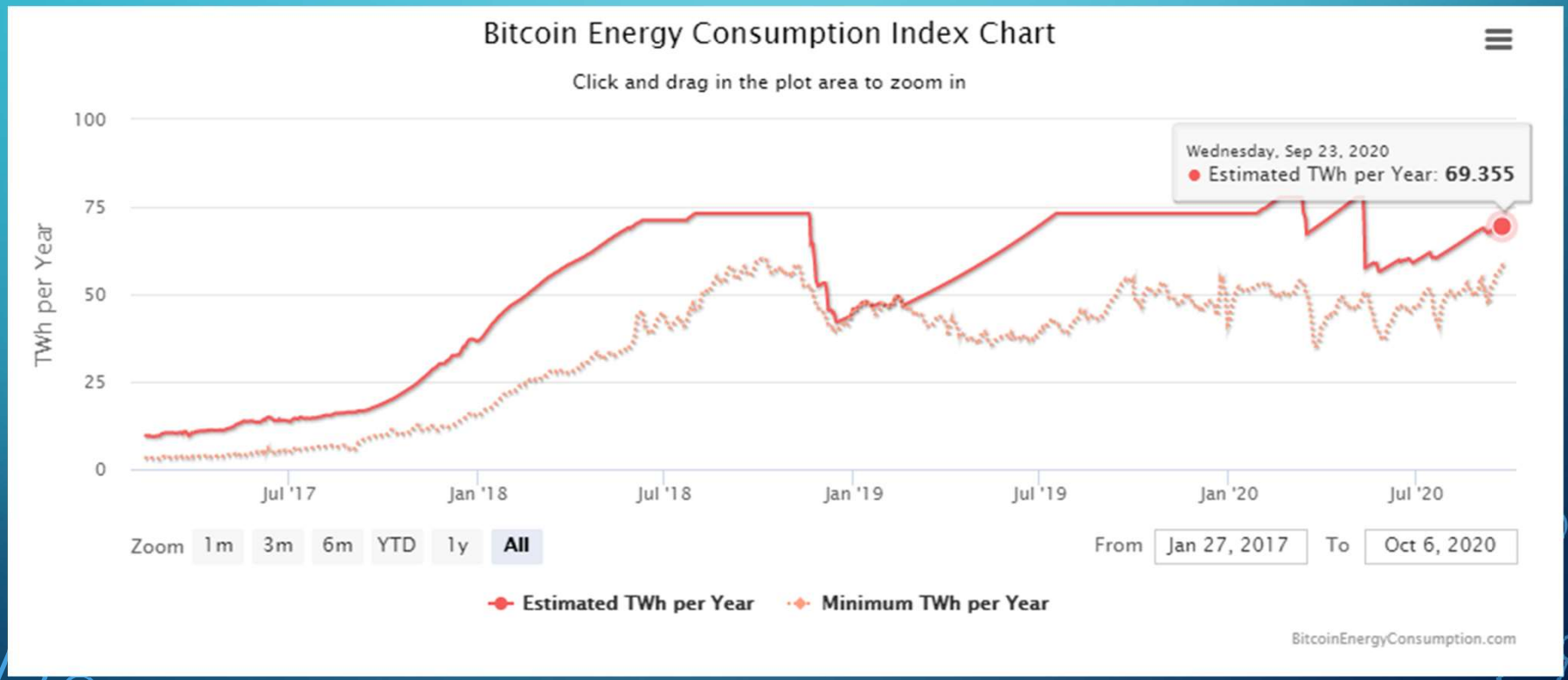
WHAT IS BLOCKCHAIN?

DEMO

https://anders.com/blockchain/

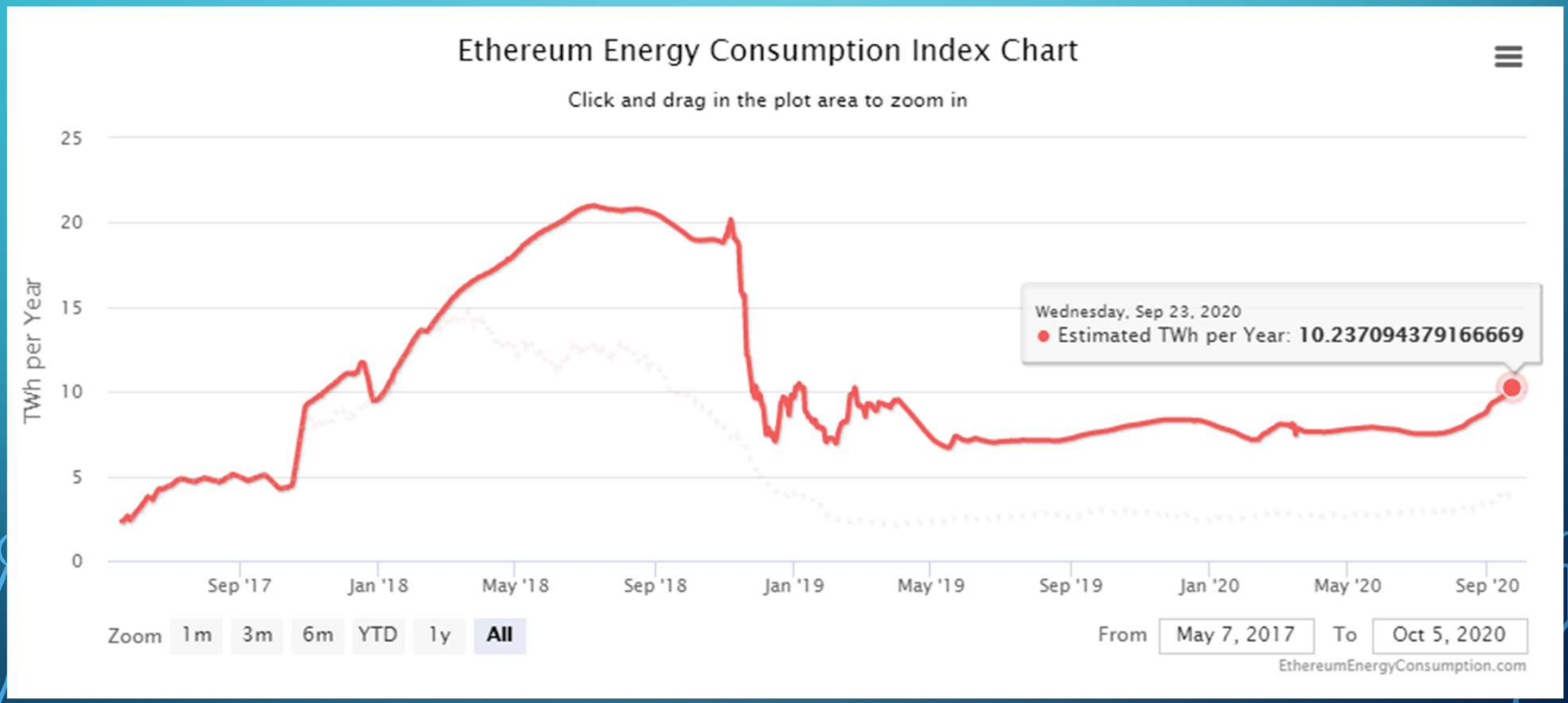
WHAT IS BLOCKCHAIN?

- Isn't proof-of work costly... ?



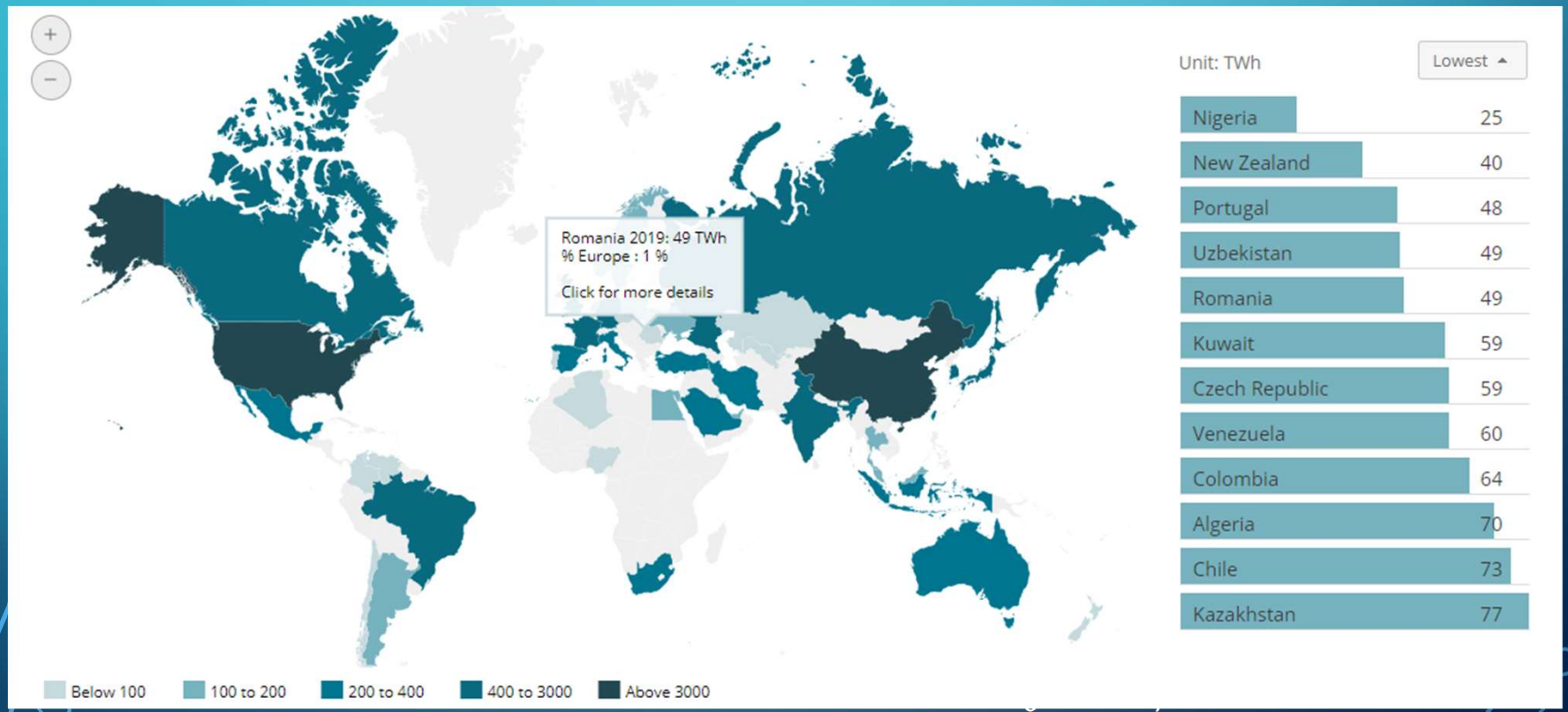
WHAT IS BLOCKCHAIN?

- Isn't proof-of work costly... ?



WHAT IS BLOCKCHAIN?

- Isn't proof-of work costly... ?



WHAT IS BLOCKCHAIN?

- The proof-of work is costly. But it's a simple & safe solution.

- **In essence,** PoW applies to *permissionless* (also referred as *public*) blockchain architectures where peers can freely join and leave

- Alternatives to proof-of-work:
  - proof-of-stake
  - proof-of-space/time
  - proof-of-authority
  - and others

- More to be discussed in future lectures, as well as particular cases for all the „**in essence**" parts

WHAT IS BLOCKCHAIN?

- *Permissioned* blockchain architectures:
  - An access control layer is implemented to permit access to the blockchain:
  - Knowledge of all participating peers is available at all times
  - This typically implies some level of centralization
  - Injecting fake peers is not possible, therefore PoW is typically not needed
  - Malicious peers can still try to tamper blocks and consensus is still needed
  - More economic BFT-tolerant measures can be implemented
  - However, typically these do not scale well on large numbers of peers
- More also to be discussed in future lectures

APPLICATION EXAMPLES

- We mostly discussed blockchain from the perspective of a cryptocurrency – i.e., using transactions to send money
- However, a record in the ledger could also hold something else and a transaction could express also other interaction between peers
- The great advantage of blockchain:

  It provides a decentralized common base of trust and immutability
- So, what could that be useful for?

APPLICATION EXAMPLES

- Too many use cases to count:

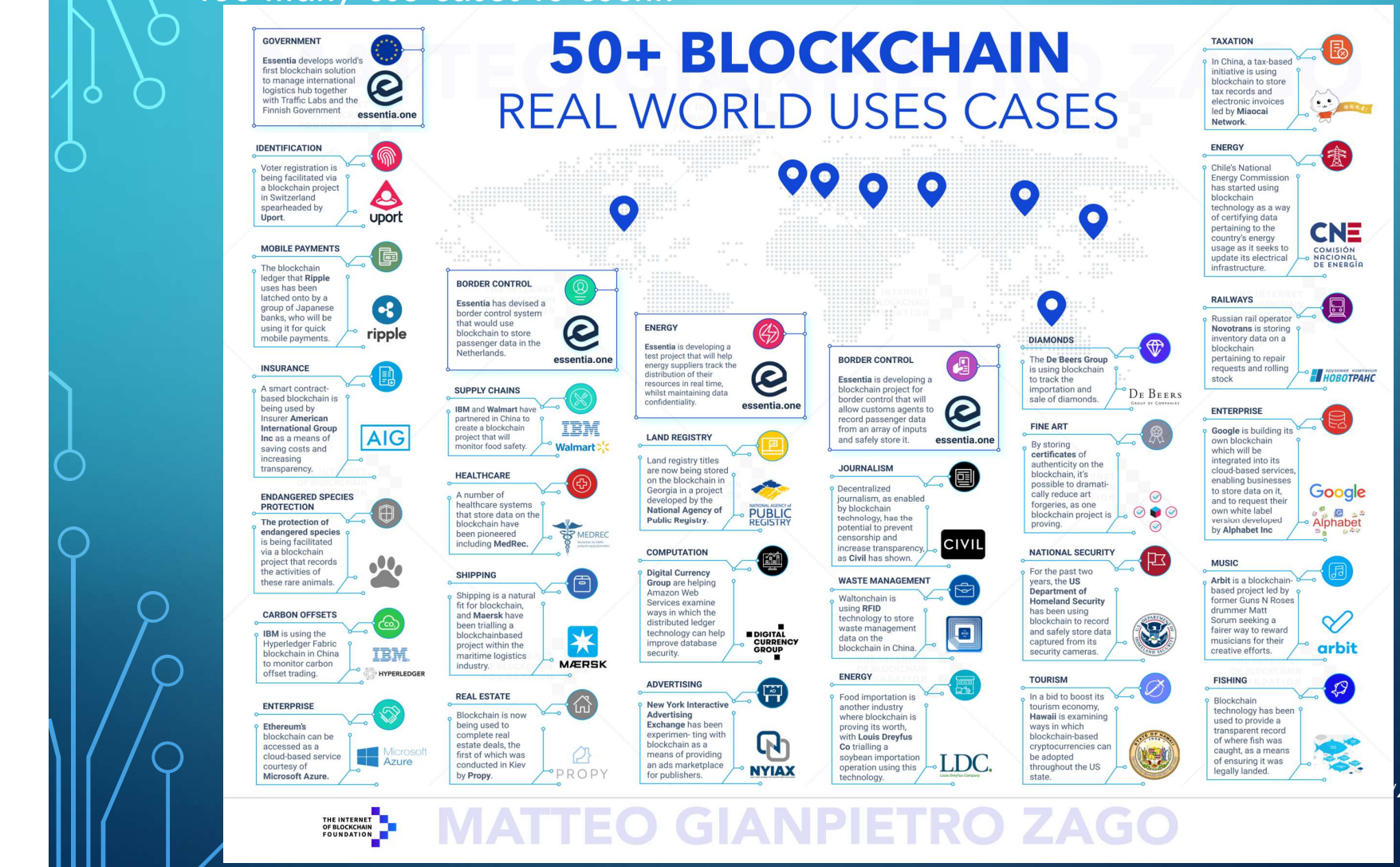


Figure source: *medium.com* (M.G.Zago), 2018

APPLICATION EXAMPLES

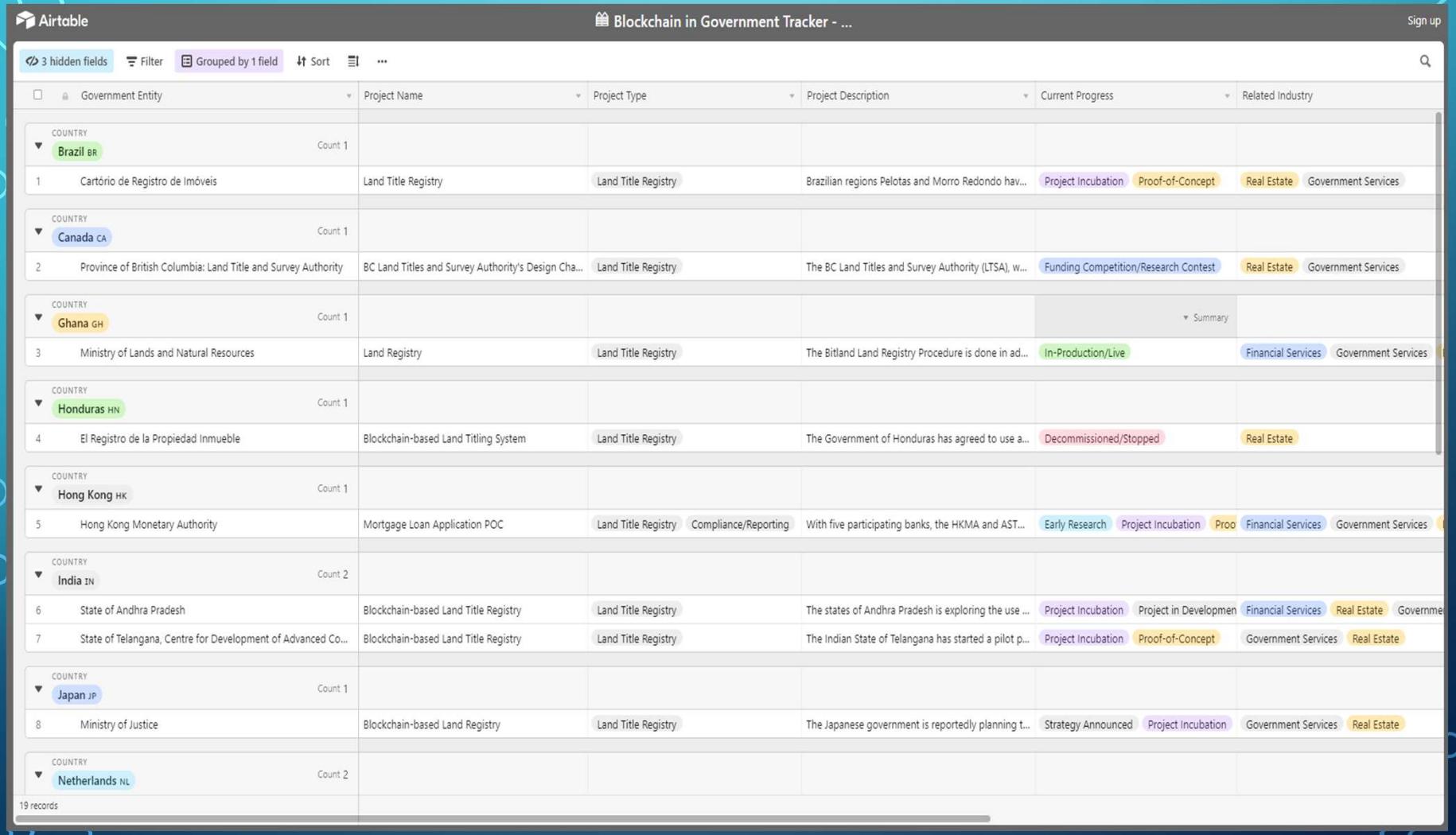
- FinTech:
  - Currency transactions (we already talked about this)
  - Insurance claims and settlements
  - Trading stocks, derivatives or other investment transactions
  - Crowdfunding
  - many others

- Internet-of-Things:
  - Smart locks (renting objects without centralized payment – slock.it)
  - Smart parking (using smart meters without centralized payment)
  - Smart applications for connected vehicles
  - many other smart stuff

APPLICATION EXAMPLES

- Industry and manufacturing:
  - Machine maintenance and diagnostics (e.g., transactions for part or consumables replacement)
  - Supply chain tracking (registry of products and tracking their possession while transferred)
  - Supplier reputation tracking (e.g., issuing seller ratings via transactions when deliveries are confirmed)
  - many others
- Records, identities and healthcare:
  - Automobile records (tracking multiple owners)
  - Land registry (tracking multiple owners)
  - Electronic health records (tracking patient's history)
  - many others

APPLICATION EXAMPLES



	Government Entity	Project Name	Project Type	Project Description	Current Progress	Related Industry
COUNTRY	Brazil BR	Count 1				
1	Cartório de Registro de Imóveis	Land Title Registry	Land Title Registry	Brazilian regions Pelotas and Morro Redondo hav...	Project Incubation, Proof-of-Concept	Real Estate, Government Services
COUNTRY	Canada CA	Count 1				
2	Province of British Columbia: Land Title and Survey Authority	BC Land Titles and Survey Authority's Design Cha...	Land Title Registry	The BC Land Titles and Survey Authority (LTSA), w...	Funding Competition/Research Contest	Real Estate, Government Services
COUNTRY	Ghana GH	Count 1			Summary	
3	Ministry of Lands and Natural Resources	Land Registry	Land Title Registry	The Bitland Land Registry Procedure is done in ad...	In-Production/Live	Financial Services, Government Services
COUNTRY	Honduras HN	Count 1				
4	El Registro de la Propiedad Inmueble	Blockchain-based Land Titling System	Land Title Registry	The Government of Honduras has agreed to use a...	Decommissioned/Stopped	Real Estate
COUNTRY	Hong Kong HK	Count 1				
5	Hong Kong Monetary Authority	Mortgage Loan Application POC	Land Title Registry, Compliance/Reporting	With five participating banks, the HKMA and AST...	Early Research, Project Incubation, Proo	Financial Services, Government Services
COUNTRY	India IN	Count 2				
6	State of Andhra Pradesh	Blockchain-based Land Title Registry	Land Title Registry	The states of Andhra Pradesh is exploring the use ...	Project Incubation, Project in Developmen	Financial Services, Real Estate, Governme
7	State of Telangana, Centre for Development of Advanced Co...	Blockchain-based Land Title Registry	Land Title Registry	The Indian State of Telangana has started a pilot p...	Project Incubation, Proof-of-Concept	Government Services, Real Estate
COUNTRY	Japan JP	Count 1				
8	Ministry of Justice	Blockchain-based Land Registry	Land Title Registry	The Japanese government is reportedly planning t...	Strategy Announced, Project Incubation	Government Services, Real Estate
COUNTRY	Netherlands NL	Count 2				

19 records

APPLICATION EXAMPLES

Figure source: *finder.com.au*

Figure source: *cointelegraph.com*

Figure source: *reuters.com*

Figure source: *cryptoslate.com*

- Many use cases rely on *smart contracts* (we'll discuss these in a future course)
- **Note:** all these and previous are examples of how blockchain *is used* (or *suggested to be used*) not necessarily on how blockchain *should be used*

BLOCKCHAIN PLATFORMS SUMMARY

- What blockchain platforms are out there?

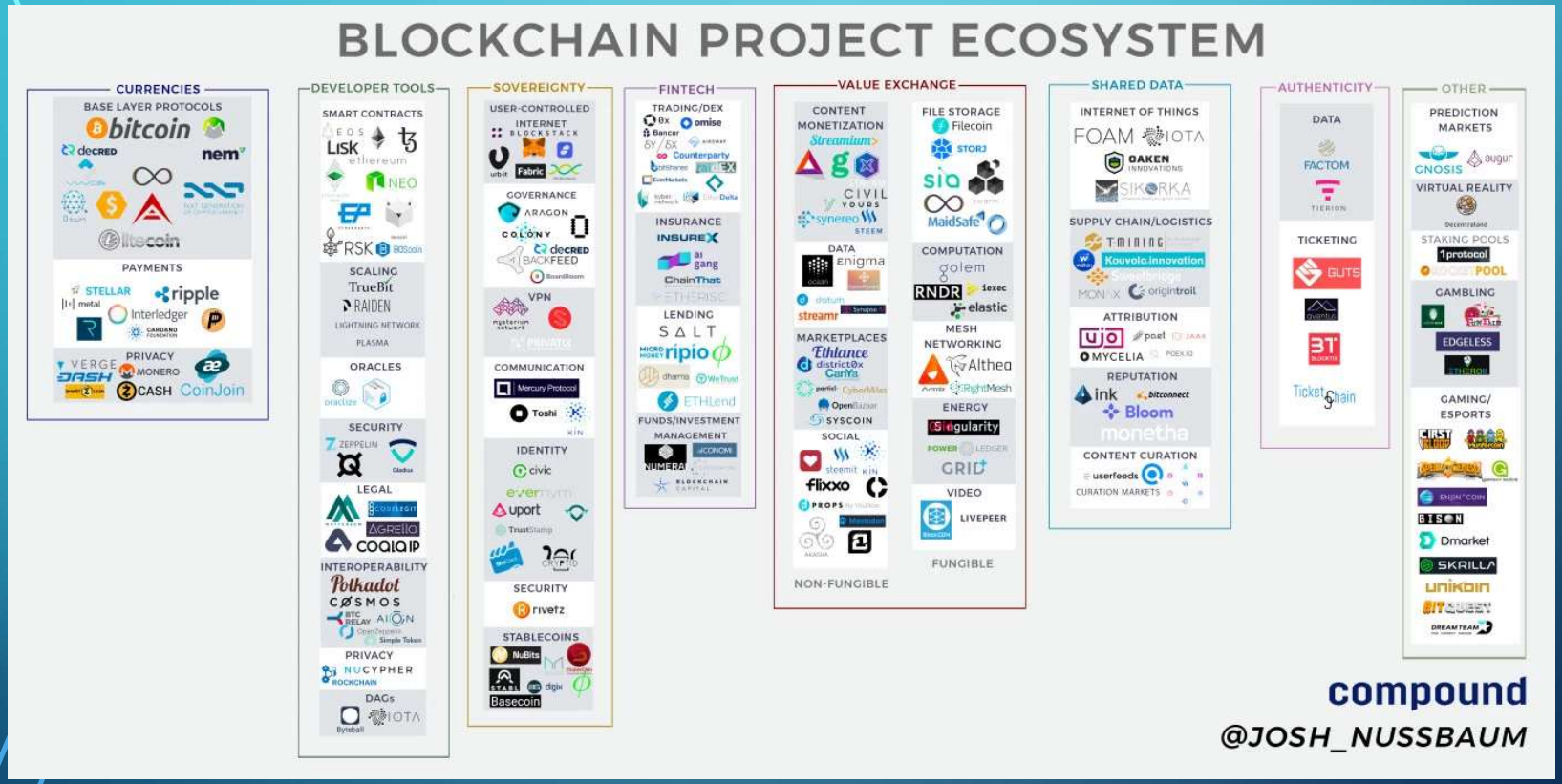


Figure source: *techcrunch.com* (J. Nubaum), 2017

- Yet again, too many to count...

BLOCKCHAIN PLATFORMS SUMMARY

- Probably the current three most representative:

- Cryptocurrencies – permissionless: **Bitcoin**
  - We already covered a bit of this
  - More in a future lecture

- Various apps – permissionless: **Ethereum**
  - Relies on a specific language (Solidity)
  - Proof-of-work based (to switch to proof-of-stake)
  - Has also a currency attached – ETH
  - The focus for the first weeks of this course

- Various apps – permissioned: **Hyperledger Fabric**
  - Uses common languages (Go, JS)
  - Part of a blockchain umbrella of platforms
  - Developed by Linux Foundation
  - We'll cover this in some later course

WHAT ISN'T BLOCKCHAIN?

- Remember the **note** a couple slides ago on how blockchain should be used ;) :

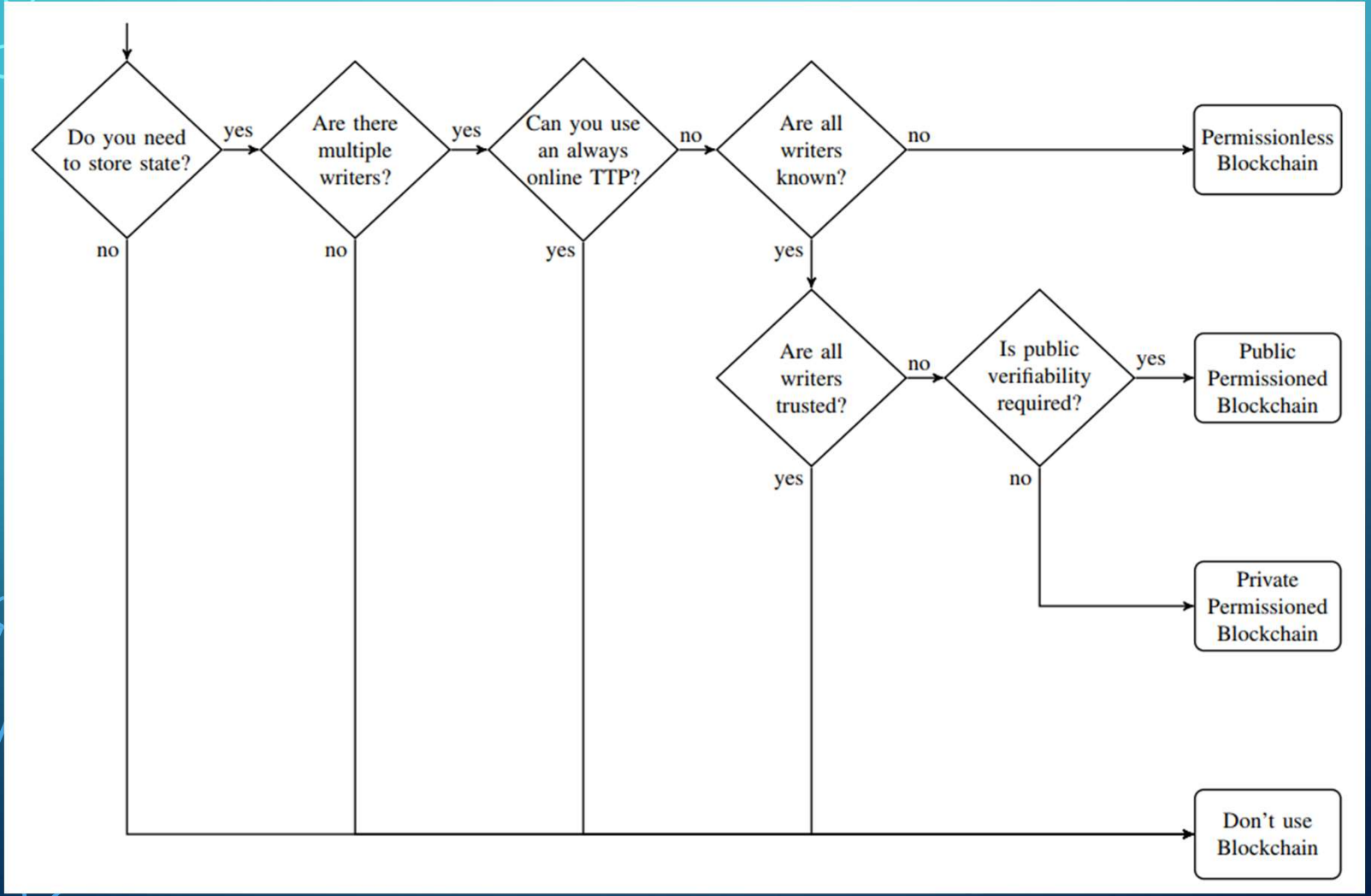


Figure source: *Do You Need a Blockchain* (K. Wüst, A. Gervais), 2017

WHAT ISN'T BLOCKCHAIN?

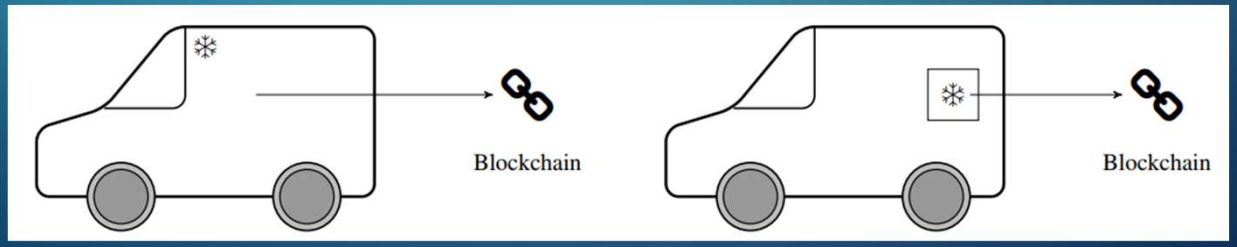
- There's even a „live demo" for this ;) :

http://doyouneedablockchain.com/

- But the „most probably" part is debatable!

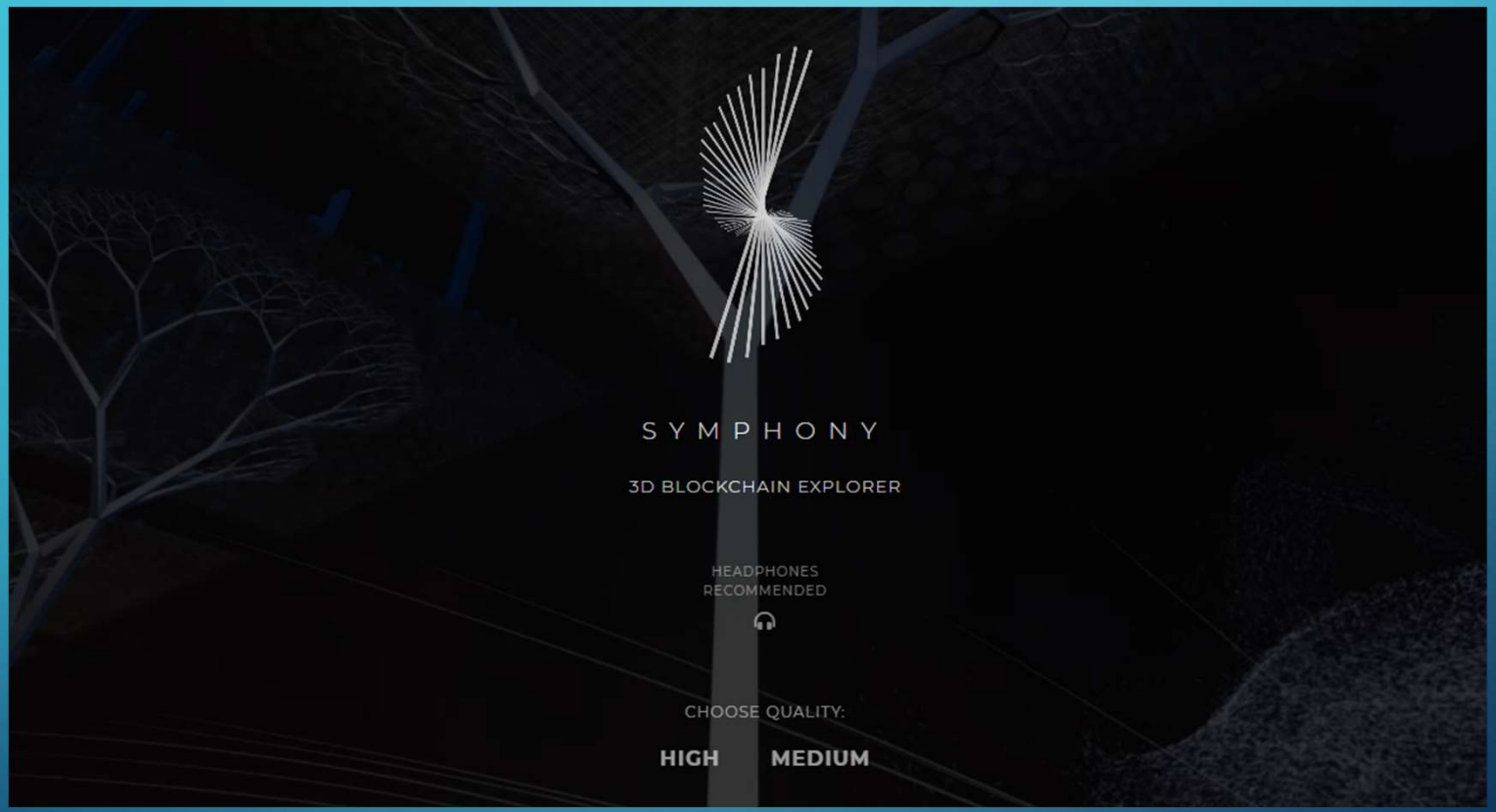
WHAT ISN'T BLOCKCHAIN?

- Many cases of blockchain overuse just due to hype/marketing purpose
- Blockchain isn't a *distributed database:*
  - often enough for what is needed
  - better performance and scalability
  - examples: Cassandra, HBase, MongoDB, others
- Blockchain isn't a „smart" solution by default. Example:
  - the new smart IoT-supply-chain-management ACME startup collects distributed sensor measurements in a blockchain for immutable tamper proof monitoring of suppliers
  - a user of the system can hold his suppliers accountable for improper transportation
  - a freezer truck use case: intended vs. applied (= cheaper to cool a fridge in the truck)



Example and figure source: *Do You Need a Blockchain* (K. Wüst, A. Gervais), 2017

EXPLORE THE BITCOIN BLOCKCHAIN



https://symphony.iohk.io/