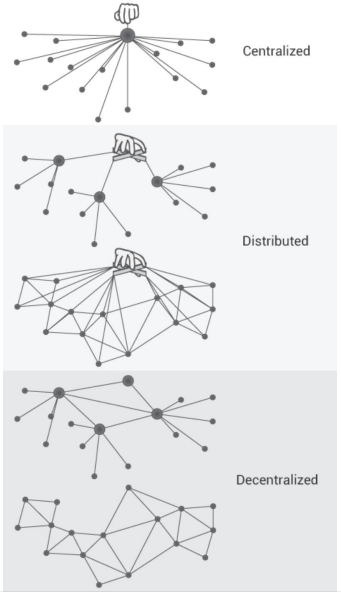
Blockchain and IoT

Andrea Vitaletti



Transaction

An atomic event supported by the protocol

- Bob → Alice 1 cryptocurrency (e.g. BTC)
- State 1 → State 2 in a decentralised computer

What is blockchain?

from trust to consensus

A decentralised ledger that can record transactions between two parties “efficiently” and in a verifiable and permanent way

What transactions are recorded and in which order?

- ❏ permissioned
- ❏ permissionless

Who can read the transaction written in chain?

- ❏ private
- ❏ public

TECHNOLOGIES OF A BLOCKCHAIN

Asymmetric Encryption
Transaction signing

Hash Functions
Transaction/block hashing as well as obfuscating public keys

Merkle Trees
Efficient way to package transactions into blocks

Key-Value Database
Lookups of previous transactions (prevent double-spends)

P2P Communication Protocol
Sharing transactions and blocks

Proof of Work
Method to achieve consensus

Technologies powering Blockchain. Source: Flaxman 2017

A --> B 1 euro

Pub_A --> Pub_B a new car <-- how to uniquely identify the new car

sign(Pub_A --> Pub_B 1) <-- you know the Sec_A

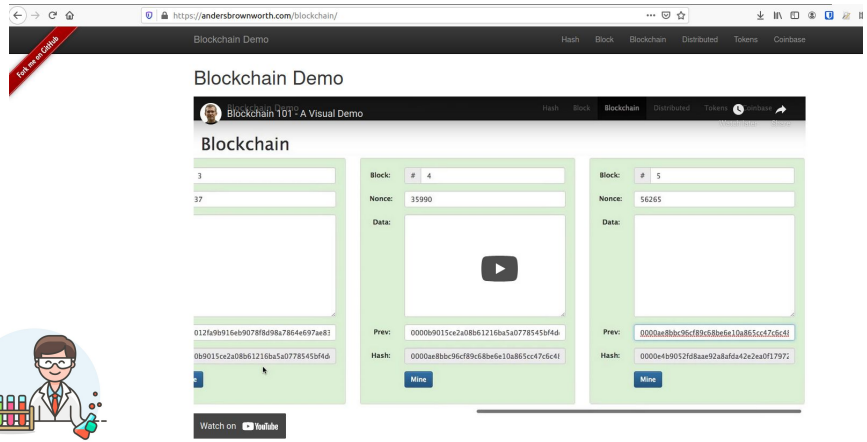
I generate a fake Pub_fake --> you don't know the associated Sec_fake

Pub_B --> Pub_fake 1

sign(Pub_B --> Pub_fake 1) <-- you know the Sec_B

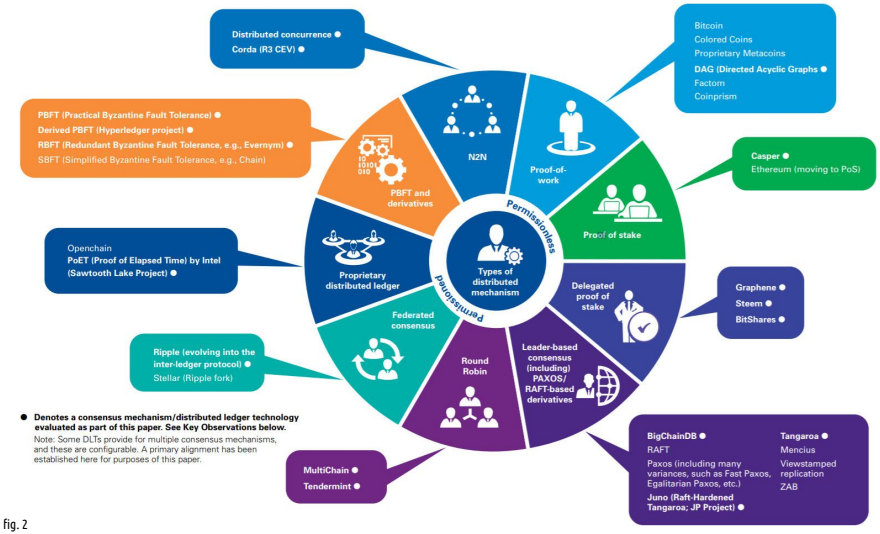
Is it possible to associate Pub_A to Andrea (legal person) ??? ← pseudo-anonimity

A nice resource

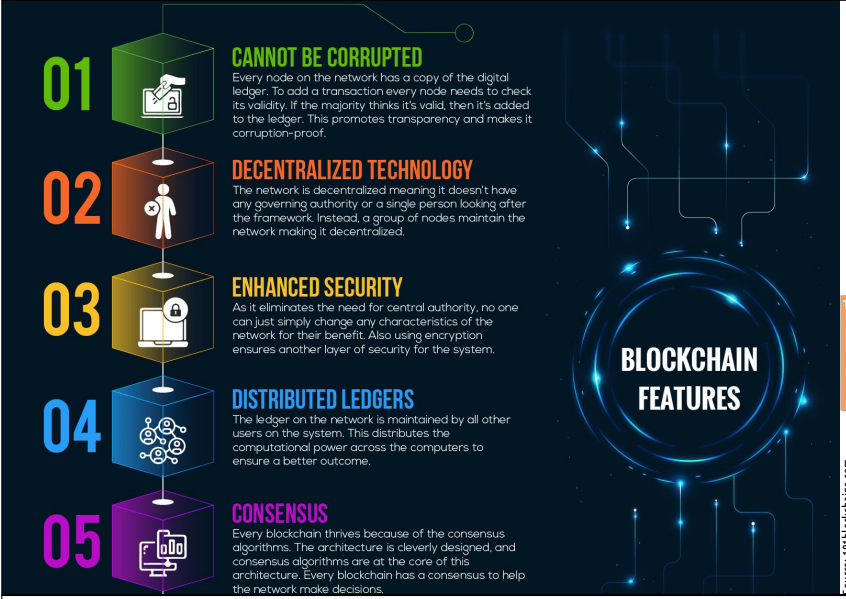


https://andersbrownworth.com/blockchain/

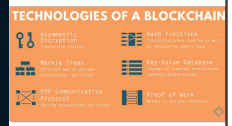
Consensus



Source: Seibold and Samman 2016, fig. 2



Source: 101blockchains.com



SAPIENZA UNIVERSITÀ DI ROMA

DB vs Blockchain

CRUD → CRAB

***Burn**: Deleting something from a blockchain conflicts with the immutability. We can stop the ability to `TRANSFER` the asset by transferring it to an un-spendable public key. We generated an artificial public key that looks like this: `BurnBurnBurnBurnBurnBurnBurnBurnBurnBurn`. The likelihood to generate a vanity address (and know the private key) that is 11 times "Burn" is extremely low.*

It is very clear that the **data itself is never deleted. Only the keys to control the transfer of data are lost** in a burn operation.

https://blog.bigchaindb.com/crab-create-retrieve-append-burn-b9f6d111f460

SAPIENZA UNIVERSITÀ DI ROMA

A --> B 1 euro

Pub_A --> Pub_B a new car <-- how to uniquely identify the new car

sign(Pub_A --> Pub_B 1) <-- you know the Sec_A

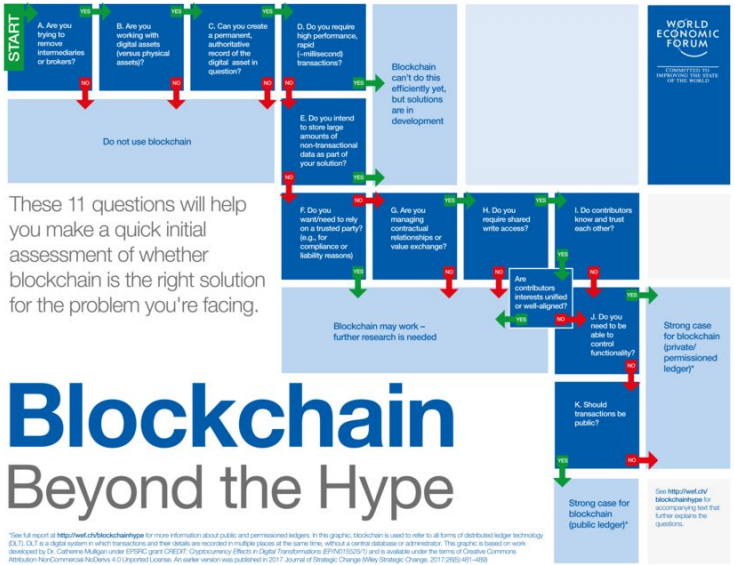
I generate a fake Pub_fake --> you don't know the associated Sec_fake

Pub_B --> Pub_fake 1

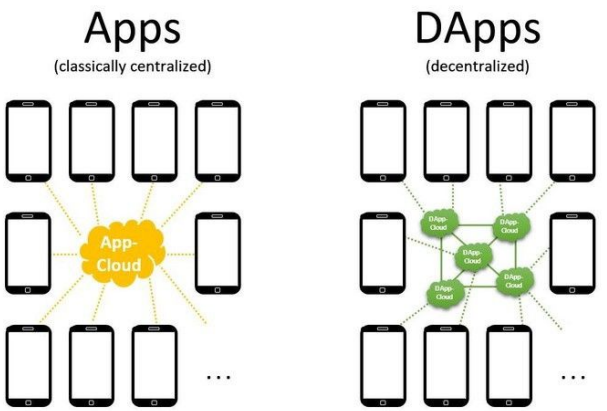
sign(Pub_B --> Pub_fake 1) <-- you know the Sec_B

Is it possible to associate Pub_A to Andrea (legal person) ??? ← pseudo-anonimity

Do We need Blockchain?



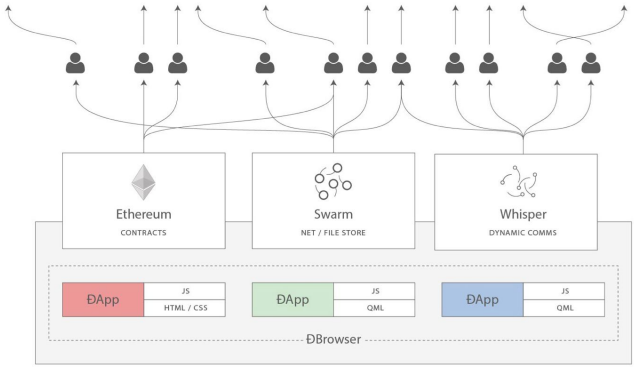
From App to Dapp the "world computer"



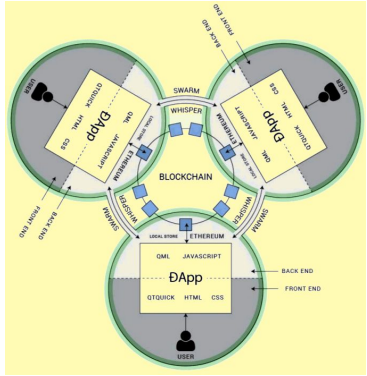
- **BEYOND C/S:**
- **DECENTRALIZED**
- **TRUSTLESS**
- **DISINTERMEDIATION**

Source: Medium

A reference architecture



Source: Héctor Ugarte on ResearchGate



Source: https://ethereum.stackexchange.com/

- Contracts: decentralized logic
- Swarm: decentralized storage
- Whisper: decentralized messaging

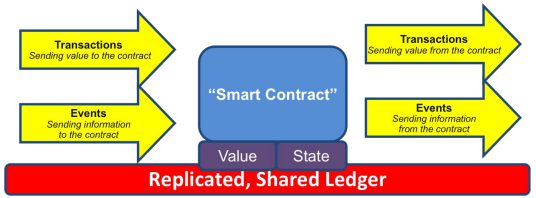
Smart Contracts

```
pragma solidity >=0.7.0 <0.9.0;

/**
 * @title Storage
 * @dev Store & retrieve value in a variable
 */
contract Storage {

    uint256 number;

    /**
     * @dev Store value in variable
     * @param num value to store
     */
    function store(uint256 num) public {
        number = num;
    }

    /**
     * @dev Return value
     * @return value of 'number'
     */
    function retrieve() public view returns (uint256){
        return number;
    }
}
```

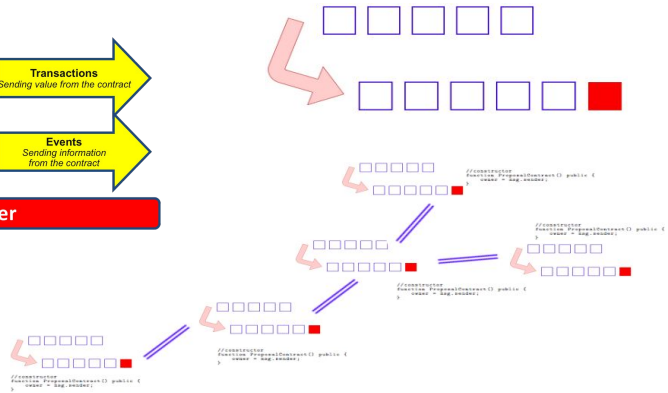
- Computer programs that may encode agreements, policies, rules and penalties that can not be arbitrarily altered once agreed and autonomously run on the blockchain.
- Transfer digital assets between parties

https://remix.ethereum.org/

The "world computer" a P2P consensus-based state machine



Source: gendal.me



Smart Contracts Platforms

Smart contract platform comparison

	Smart Contracts		Protocol Details			Other
	Turing Complete	Verifiable	Consensus Process	Interoperability	Scalability	Governance
DFINITY	Yes	No	Threshold Relay PoW/PoS	Interoperable with Ethereum	Parallell mining networks	Self amending protocol informed by Futacratic DAO
Qtum	Yes	No	UTXO PoS	Interoperable with Ethereum & Bitcoin heritage networks	Contract code executed off chain	Foundation informed by DAO
cosmos	No (Can trigger contracts on other chains)	No	Federated PoS	Interoperable with all networks connected through a hub chain	Paralell side chains & interconnected blockchains	DAO
BOScoin	No	No	Federated PoS	Minimal Interoperability	Small cohort of semitrusted nodes quickly reach consensus	DAO
ethereum	Yes	In development	PoW (Transitioning to Casper)	Minimal Interoperability	Sharding & possible implementation of Raiden Network	Foundation
tezos	Yes	Yes	Delegated PoS	Interoperable with Ethereum	Contract code executed off chain & paralell consensus networks	Self amending protocol informed by a DAO (possible shift to futacratic governance later)
	Yes	No	PoS/PoW	Interoperable with Ethereum	State channels	Futacratic DAO

Source: Smith + Crown table compiled DFINITY, Qtum, Cosmos, BOScoin, Ethereum, Tezos and Aeternity whitepapers and other official soruces

Source: Alina Nuzhnenko on Medium

PUBLIC (PERMISSIONLESS)

ETHEREUM, EOS, LISK, RADIX, UBIQ, ETHEREUM CLASSIC, NEO, TEZOS, CALYSTO, QTUM, BITSHARES, NXT, WAVES, DFINITY, NEBLIO, EXPANSE, OMNI LAYER, BYTEBALL, POA NETWORK, COUNTERPARTY, URBIT, ROOTSTOCK

PRIVATE (PERMISSIONED)

UNIVERSA, KOMODO, CORDA, HYPERLEDGER, NEM, CIPHER, MONAX, QUORUM, MIJIN, EXONUM, WANCHAIN, AETERNITY

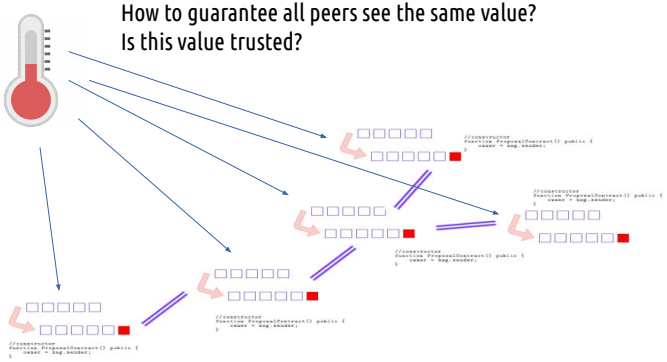
Source: Twitter

EITHER PRIVATE OR PUBLIC

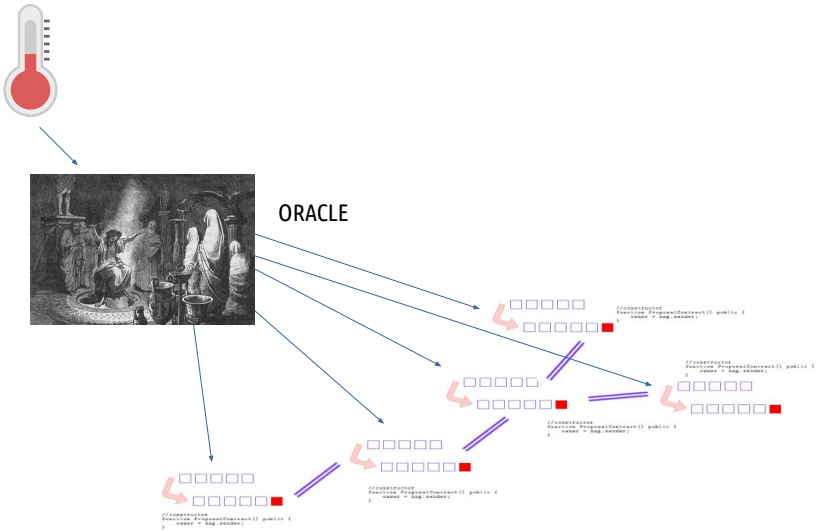
SEELE, MATRIX, WANCHAIN, APLA, CARDANO

off-chain information ← IoT data

SAPIENZA UNIVERSITÀ DI ROMA



SAPIENZA UNIVERSITÀ DI ROMA



BLOCKCHAIN ORACLES

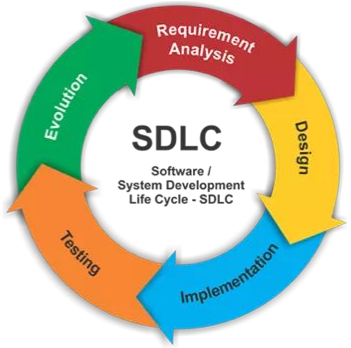
An oracle is an intermediary between blockchain and the offline world. Since a blockchain network cannot independently gather information from outside sources, an oracle is needed to relay such information

TYPES OF ORACLE

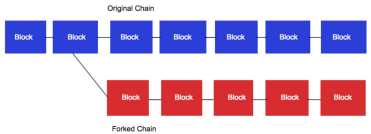
SOFTWARE	HARDWARE	INBOUND	OUTBOUND	CONSENSUS OF ORACLES
Handles data sourced online	Relays offline data from physical world	Passes external data to smart contracts	Communicates smart contract data to outside world	Increased data validity via confirmation from several oracles

https://blog.apla.io/what-is-a-blockchain-oracle-2ccca433c026

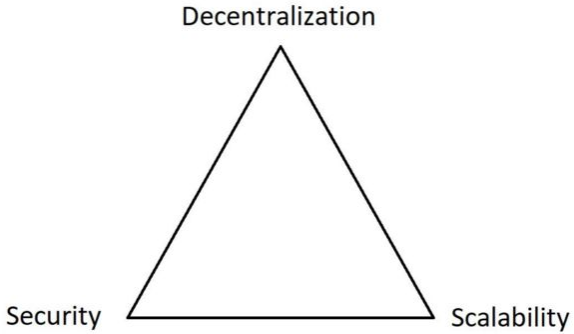
Blockchain Oriented Software Engineering (BOSE)



- **The DAO story**
- A concrete attempt to implement a funding platform, similar to Kickstarter
- Went live in 2016 with between 10-20 thousand investors (estimation) providing the equivalent of about US$ 250 million in funding and thus breaking all existing crowdfunding records.
- However, after few months an unintended behavior of the DAOs code was exploited draining the fund of millions of dollars worth of ETH tokens.



Blockchain Trilemma

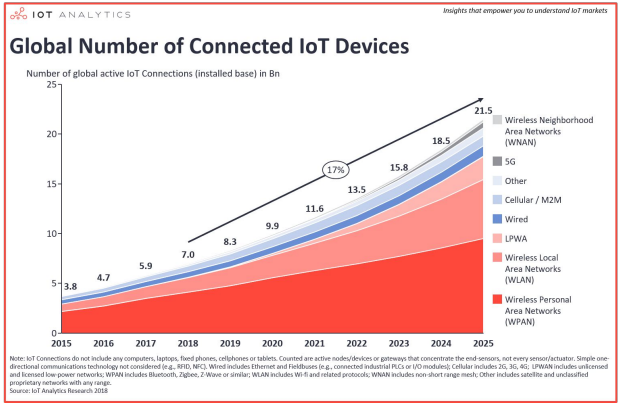


1. Decentralized: creating a blockchain system that does not rely on a central point of control
2. Scalable: the ability for a blockchain system to handle an increasingly growing amount of transactions
3. Secure: the ability of the blockchain system to operate as expected, defend itself from attacks, bugs, and other unforeseen issues

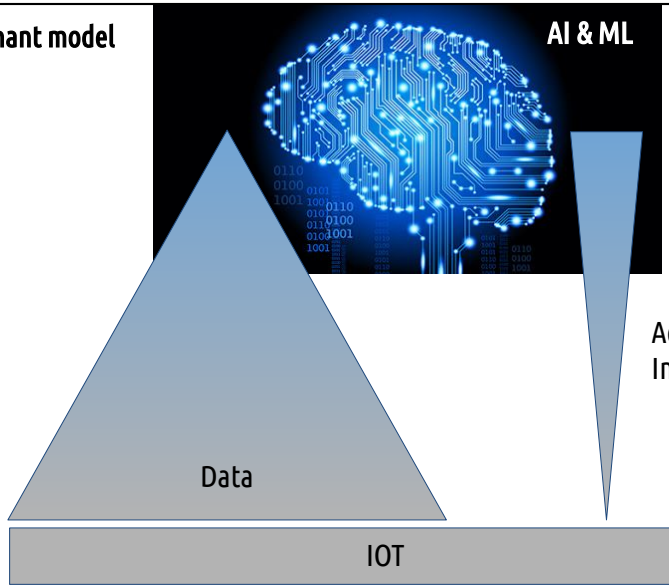
Vitalik Buterin outlined that “Blockchain systems have to trade-off between different properties. And it’s very hard for them to have three things at the same time, where one of them is decentralization. The other is scalability, and the third is security”.

IoT

- Decentralised in nature
- A huge number of devices belonging to a number of entities generate an unprecedented amount of data

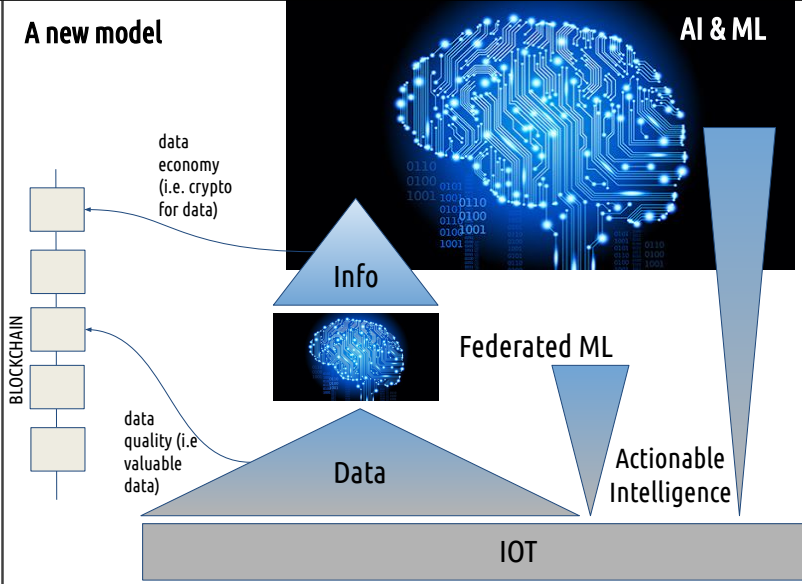


The dominant model



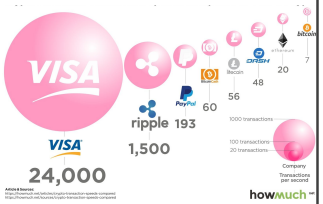
Blockchain and IoT
the big picture

A new model



Blockchain + IoT

- Data Quality: Integrity, immutability, ordering, authenticity
- Smart Contracts
- Data Economy
- Scalability
- Bandwidth
- Decentralization
  - no single point of failure
  - high level of security, but what level of weakness (if any) do the IoT devices create at the point where they connect to the network? Devices themselves will have to be secured as well to prevent hackers from tampering with them.
- Interoperability: Cross-chain interoperability will have to be addressed and improved if we truly want to leverage the benefits of interconnected smart devices. If not, we can end up with a situation where we are connected to multiple isolated decentralized networks that work well for their purpose but can't necessarily talk to other devices for which they were not specifically designed.
- Legal, compliance and regulation: The allocation of responsibility will have to be closely examined. How smart contract actions are regulated in the world outside of blockchain will also have to be stipulated. For example, who takes responsibility if an IoT-connected medical device implanted in a patient takes an action based on certain smart contract rules but ends up causing the patient harm? Is this the responsibility of the manufacturer or the IoT platform? If the IoT platform is blockchain-based, it will be decentralized without a controlling entity, so pinpointing an accountable party might present a problem.
- GDPR



European Parliament

Blockchain and the General Data Protection Regulation

Can distributed ledgers be squared with European data protection law?

https://www.europarl.europa.eu/RegData/etudes/STUD/2019/634445/EPRS_STU(2019)634445_EN.pdf

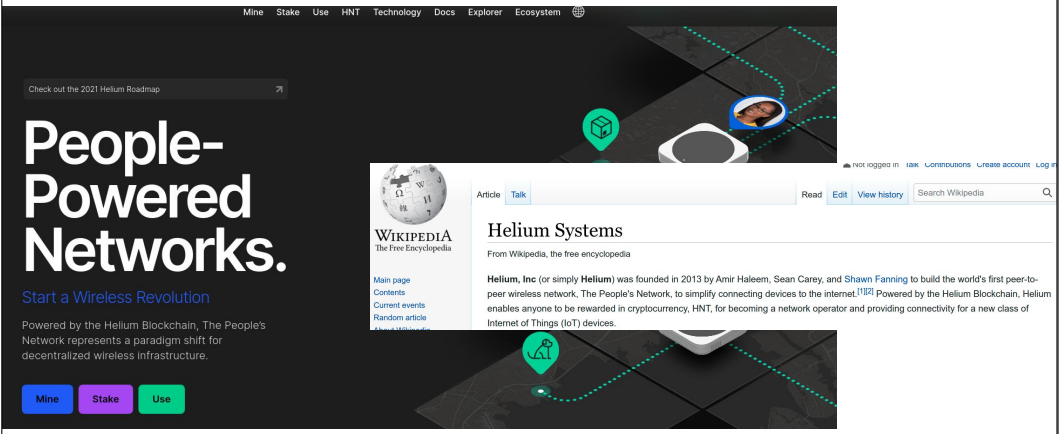
IMMUTABILITY vs RIGHT to BE FORGOTTEN

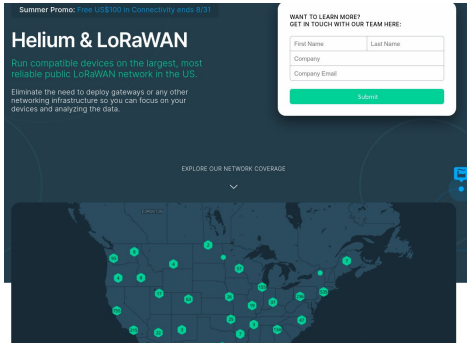
Blockchain: Challenges and solutions for compliance with the GDPR

2-model-alt-block-binary- Blockchain — TLC Jonhson

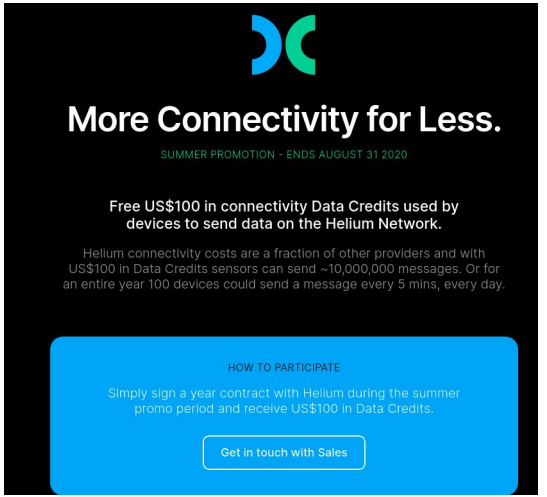
A brilliant use case

https://www.helium.com/

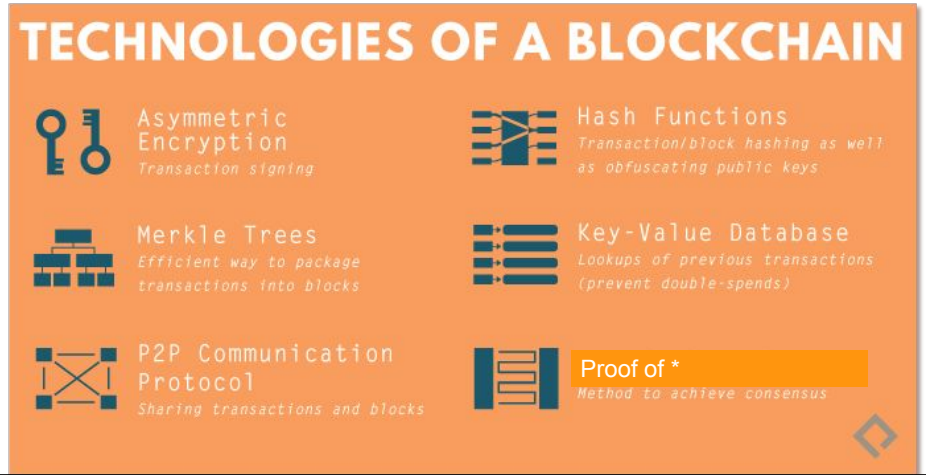




https://www.helium.com/lorawan

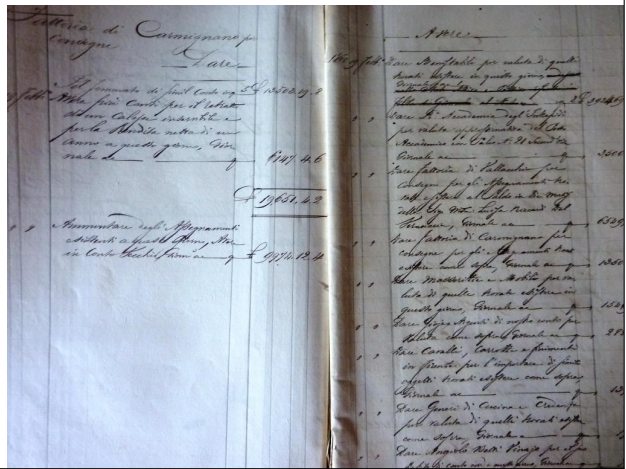


Back to the basics

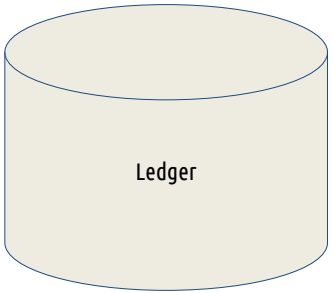


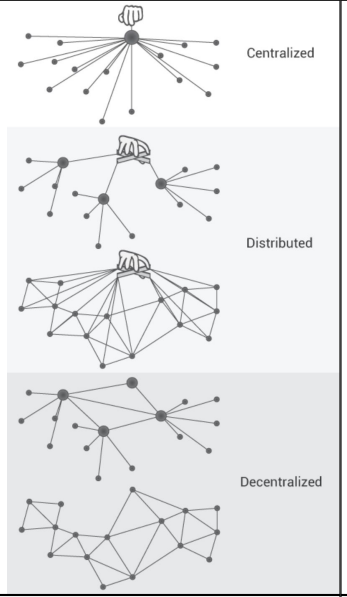
A decentralised ledger that can record transactions between two parties "efficiently" and in a verifiable and permanent way

- A decentralised ledger
- Where?

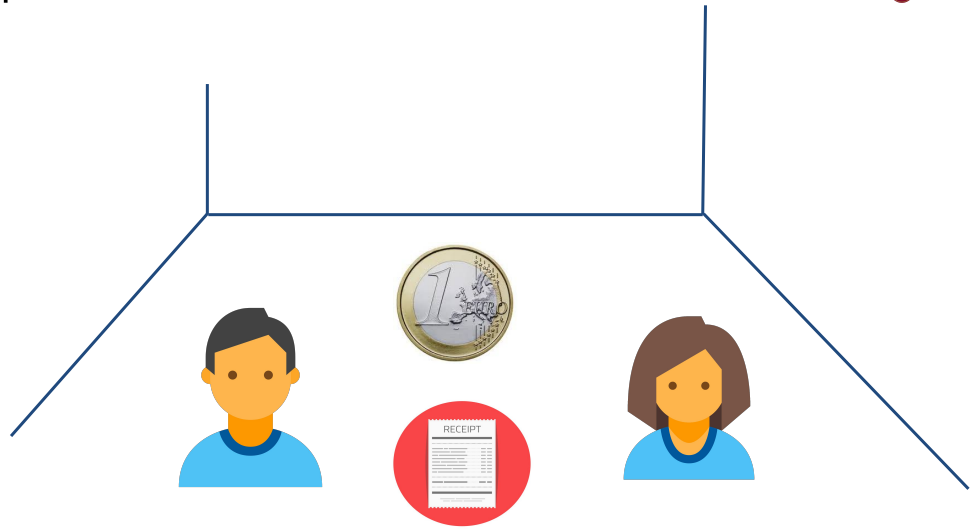


Where?

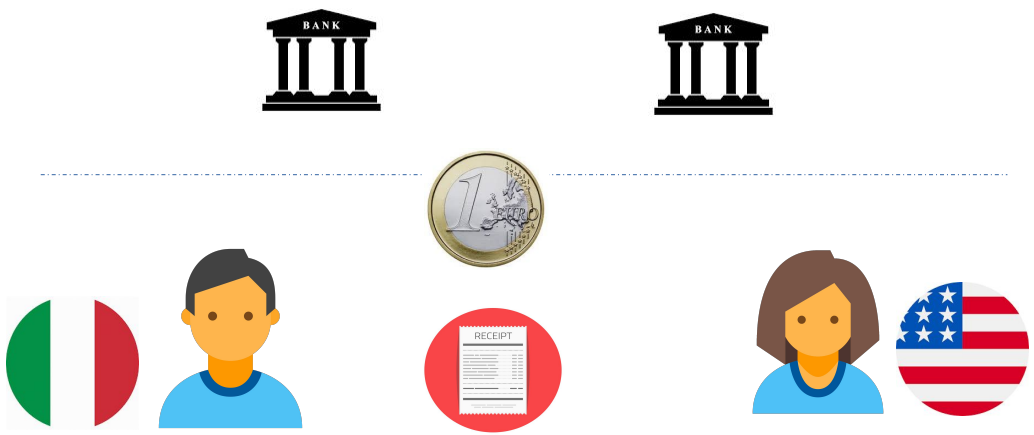




In-place: Alice → BoB 1€



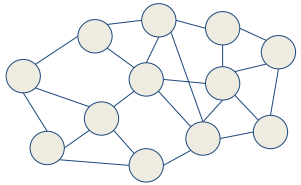
Remote/Centralised: Alice → BoB 1€

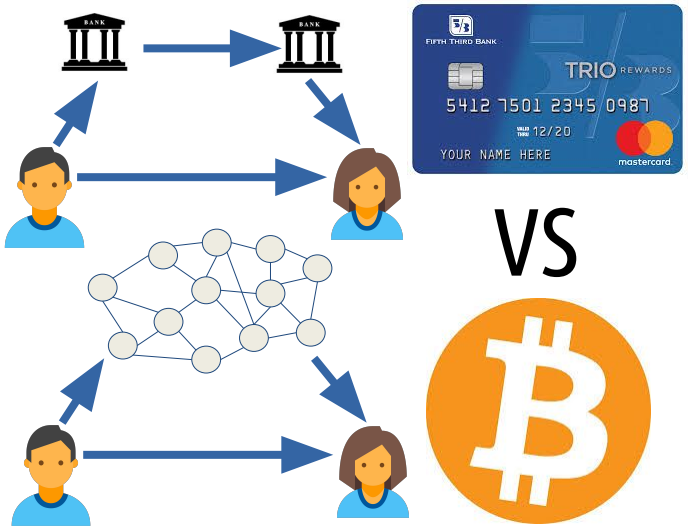


INTERMEDIARIES

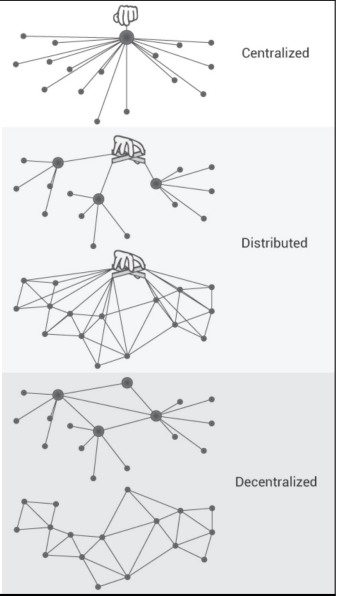
TRUST

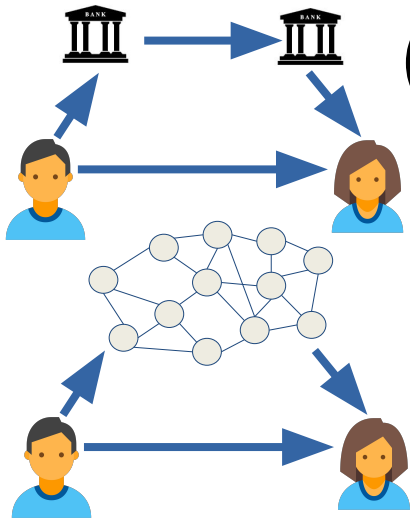
Remote/Decentralised: Alice → BoB 1€





VS

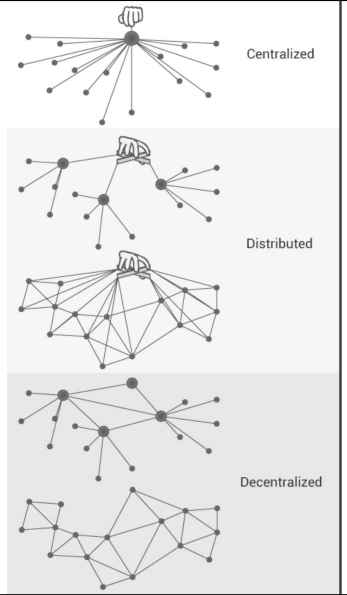




(Individuals)
TRUST

VS

CONSENSUS
(Society)



Assumption

In a society, we can expect that the majority of its member behave correctly

Bitcoin: A Peer-to-Peer Electronic Cash System

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

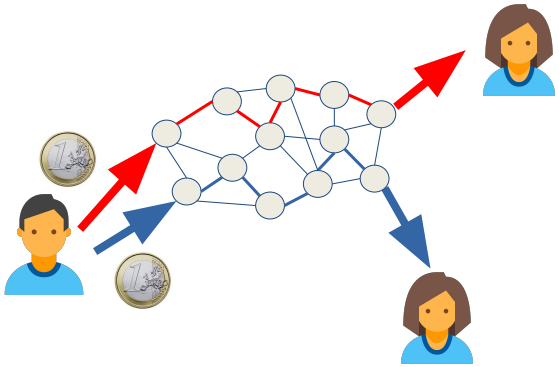
Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required 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 a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast 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.

1. Introduction

Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for most transactions, it still suffers from the inherent weaknesses of the trust based model. Completely non-reversible transactions are not really possible, since financial institutions cannot avoid mediating disputes. The cost of mediation increases transaction costs, limiting the minimum practical transaction size and cutting off the possibility for small casual transactions, and there is a broader cost in the loss of ability to make non-reversible payments for non-reversible services. With the possibility of reversal, the need for trust spreads. Merchants must be wary of their customers, hassling them for more information than they would otherwise need. A certain percentage of fraud is accepted as unavoidable. These costs and payment uncertainties can be avoided in person by using physical currency, but no mechanism exists to make payments over a communications channel without a trusted party.

10. ^ a b Nakamoto, Satoshi (24 May 2009). "Bitcoin: A Peer-to-Peer Electronic Cash System" (PDF). Archived (PDF) from the original on 20 March 2014. Retrieved 5 March 2014.
11. ^ Nakamoto, Satoshi (31 October 2008). "Bitcoin P2P e-cash paper". Archived from the original on 28 December 2012. Retrieved 5 March 2014.
12. ^ "Satoshi's posts to Cryptography mailing list". Mail-archive.com. Archived from the original on 3 January 2013. Retrieved 14 December 2013.

Order of events

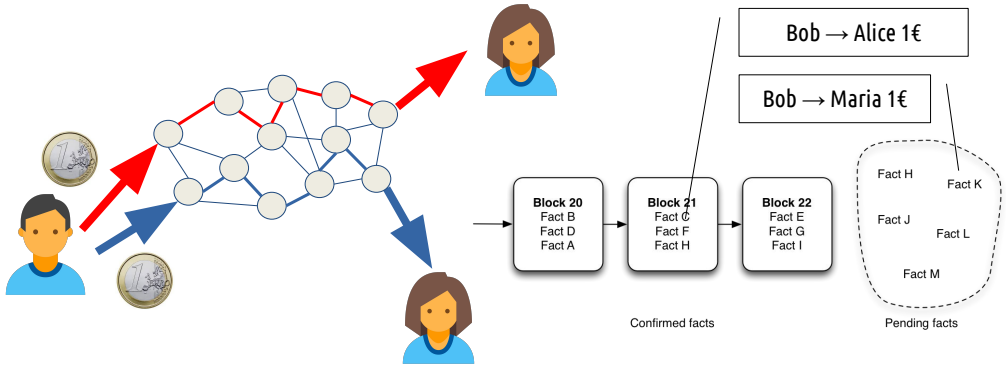


Double spending: Bob has 1€, but he spends it twice with Alice and Maria. Who will eventually get the 1€?

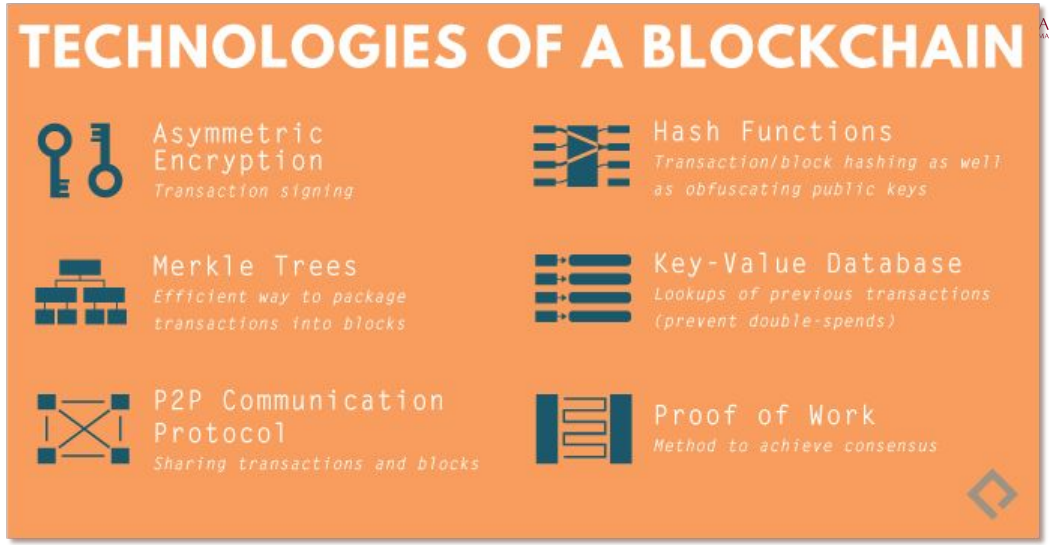
Order events (→ vs →)

If two inconciliable events happens, only the first "recorded" is considered legitimate

Order the events



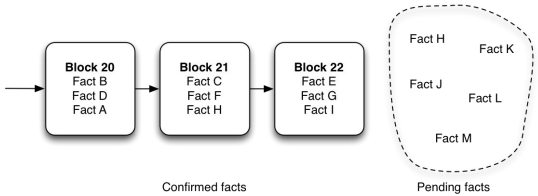
Source: https://marmelab.com/



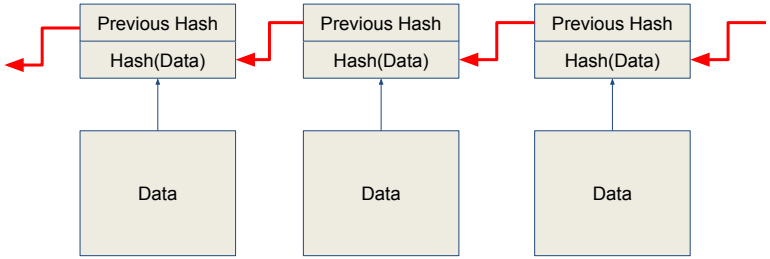
Technologies powering Blockchain. Source: Flaxman 2017

Immutability

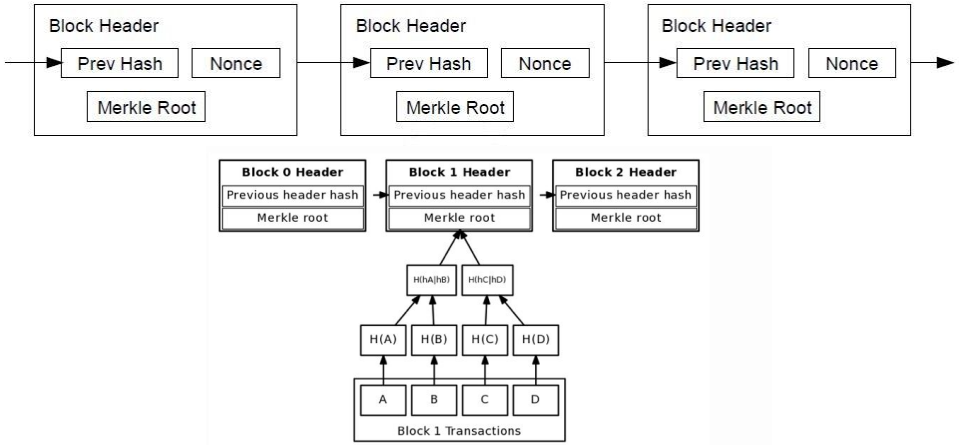
- Ordered events in the blockchain are immutable (hash)



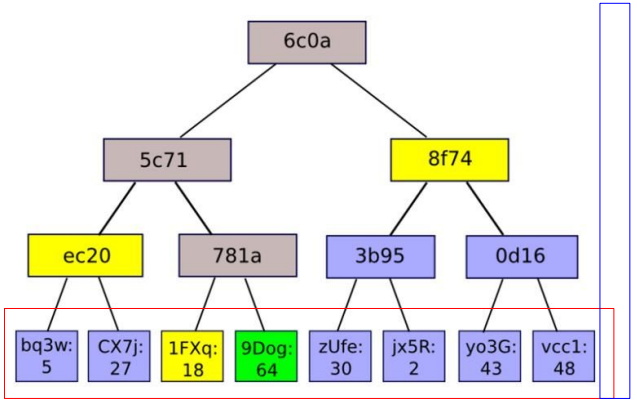
Hash and Blockchain



Hash and Blockchain



Merkle Tree: an efficient data structure to verify data



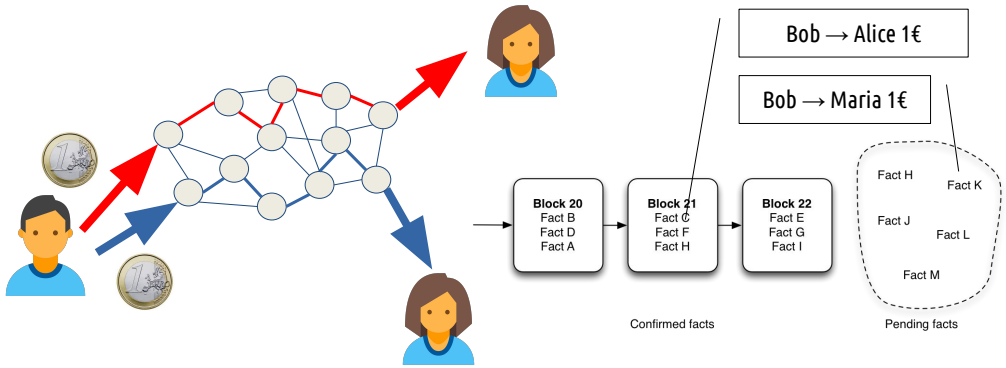
- La root del Merkle Tree è nota e fidata
- Merkle Proof:
  - Goal: la transazione verde 9Dogè nel blocco e integra
  - Proof:
    - Hash(9Dog) → 64
    - root hash → 6c0a
    - verification path: 1FXq: 18, ec20 , 8f74

hash list proof:
- n ← tutta la lista

merkle tree proof:
- log(n)

https://www.programmersought.com/article/33733422604/

Order the events



Source: https://marmelab.com/

How to select what blocks are recorded in the blockchain?

The problem of consensus

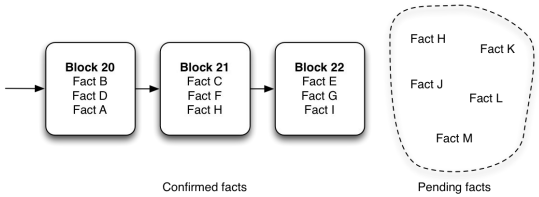
A shared agreement on the order of events

Agreement among the peers on the order of events (i.e. consensus)

The goal of consensus is to have the same exact copy of the blockchain in each node

How to add a node:

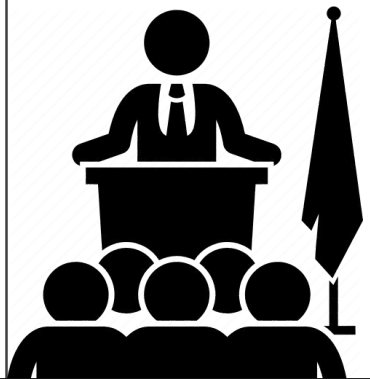
1. Propose a block as a candidate to be inserted in the blockchain
2. Reach an agreement (i.e. consensus) on the candidate block

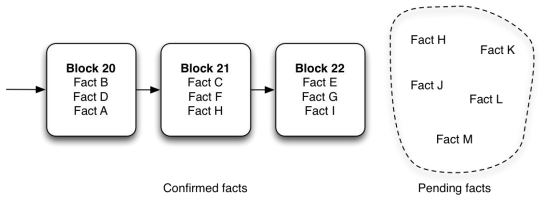


https://medium.com/cryptronics/proof-of-work-is-not-a-consensus-protocol-understanding-the-basics-of-blockchain-consensus-30aac7e845c8

Propose a block as a candidate to be inserted in the blockchain
Leader election

Elect a leader who makes a block proposal

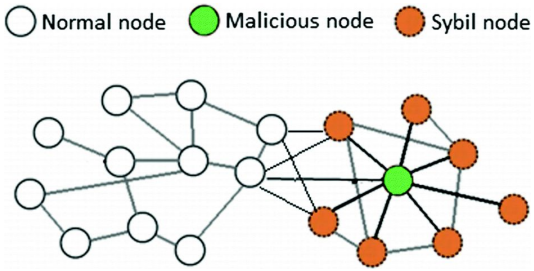




https://medium.com/loom-network/understanding-blockchain-fundamentals-part-2-proof-of-work-proof-of-stake-b6ae907c7edb

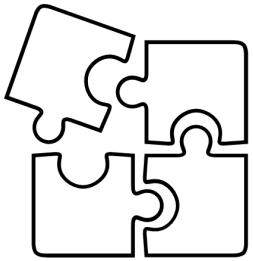
Random election

- Impersonate more nodes is easy and cheap → sybil attack
- Increase probability of winning → unfair



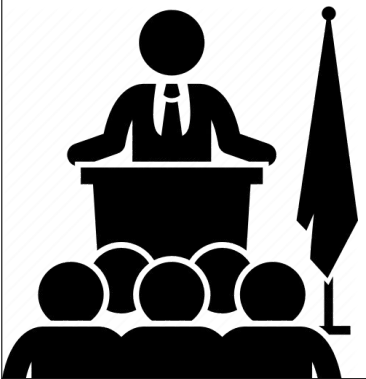
Proof-of-work

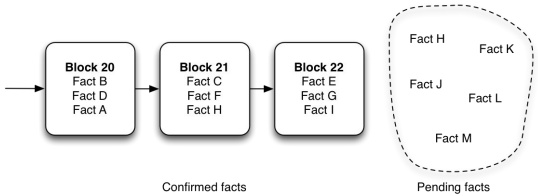
- computational power is a scarce resource
- Anti-economic (hard) acquire more computational power → difficult sybil attack
- computational power to solve a math puzzle



Leader election

Who solves the puzzle becomes leader: it proposes the block and it communicates it to peers

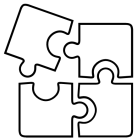


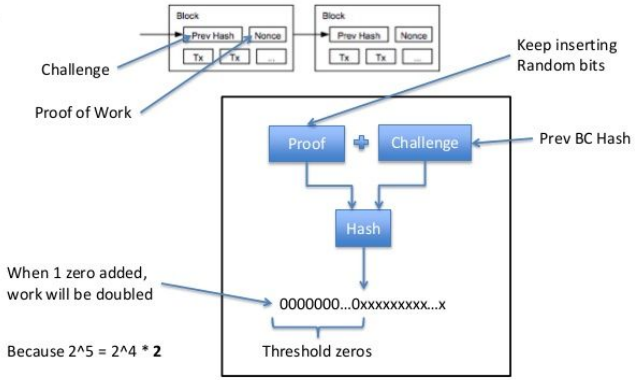


https://medium.com/loom-network/understanding-blockchain-fundamentals-part-2-proof-of-work-proof-of-stake-b6ae907c7edb

The puzzle

- No strategies to invert an hash function (i.e. Hard problem) → brute force
- Verify (by peers) is easy if you know proof and challenge





Discussion

- Brute force
  - + computational power → + probability to solve math puzzle
  - more than 50% (concentration) → win "W.h.p."
- Computational power is a scarce resource
  - concentration is difficult (easier in the RND toy example)
  - "fairness" in probability of solving the puzzle

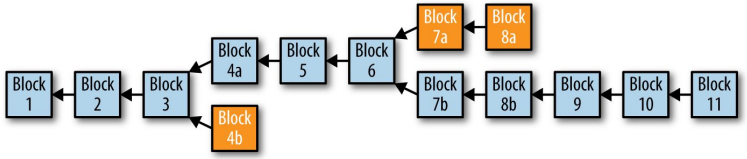
Reach an agreement (i.e. consensus) on the candidate block

Now that a candidate block has been proposed by who won the math puzzle shall we record it in chain?

Simple rules:

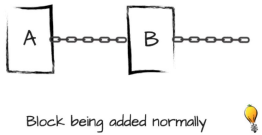
- The proposed block is valid (check by peers)
- The longest chain win → the chain supported by the majority of the computational power of the whole network of peers

Temporary Fork are possible and "quite" commons



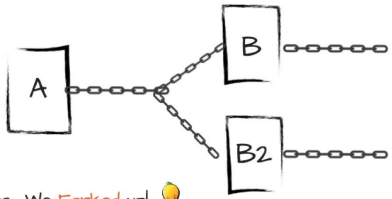
Fork

Supponiamo che l'ultimo blocco valido sia il Blocco A. I minatori ora gareggiano per il Blocco B e cercheranno di risolvere il puzzle finché non sentiranno la comunicazione di un vincitore.



Block being added normally

Tuttavia ci potrebbero essere due vincitori "simultanei".
Poiché il vincitore viene comunicato attraverso la rete, diversi partecipanti potrebbero ascoltare un vincitore diverso e dunque accettare un blocco diverso per poi passare al blocco successivo.



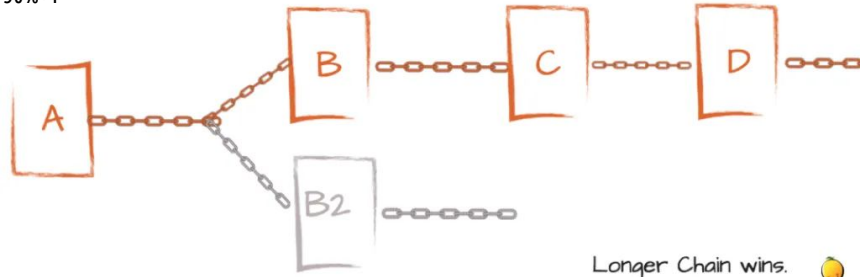
Oops.....We Forked up!

https://www.mangoresearch.co/blockchain-forks-explained/

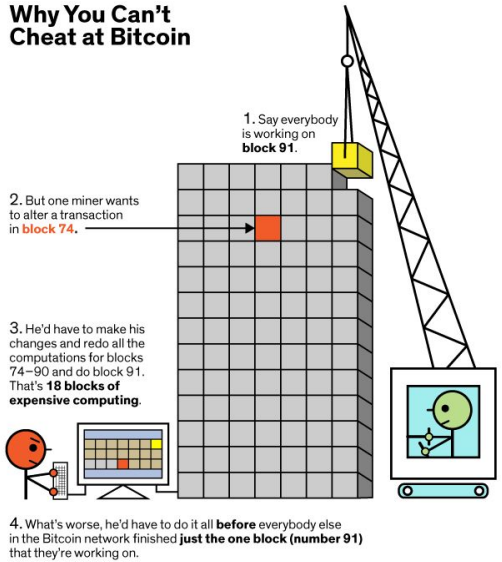
Come risolvere il fork

La chain più lunga vince → la chain supportata dalla maggioranza del potere computazionale della rete

- Potere computazionale è una risorsa scarsa
- Difficili coalizioni
- Puzzle → random winner
- 50%+1



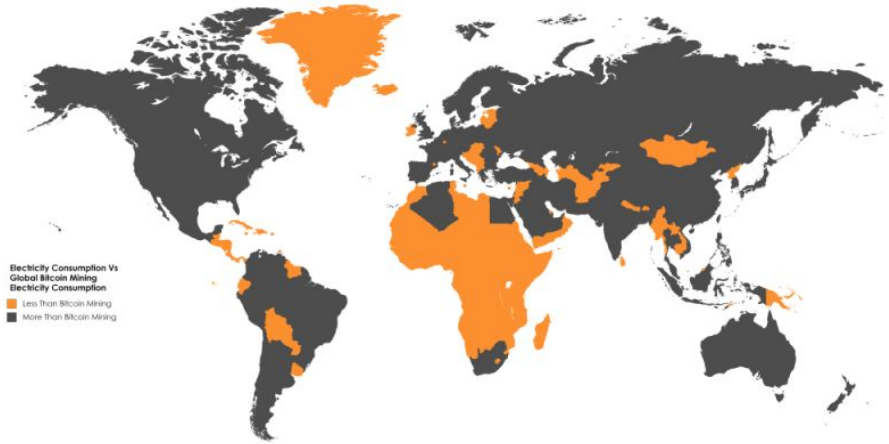
Longer Chain wins.
Shorter is "orphaned"



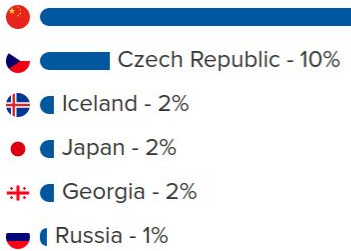
Una risorsa fantastica!

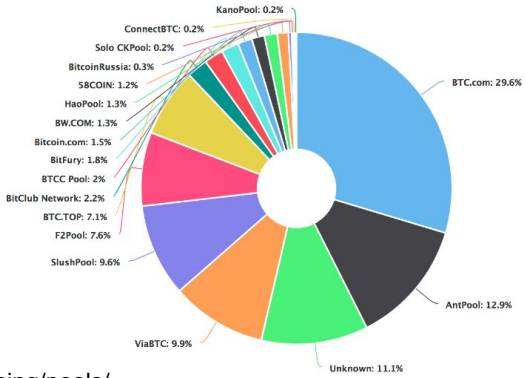
https://andersbrownworth.com/blockchain/

Sostenibilità



Concentrazione dei mining pool





https://www.buybitcoinworldwide.com/mining/pools/

Scalabilità

A more general framework

The Byzantine Generals Problem

point-to-point

LESLIE LAMPORT, ROBERT SHOSTAK, and MARSHALL PEASE
SRI International

Reliable computer systems must handle malfunctioning components that give conflicting information to different parts of the system. This situation can be expressed abstractly in terms of a group of generals of the Byzantine army camped with their troops around an enemy city. Communicating only by messenger, the generals must agree upon a common battle plan. However, one or more of them may be traitors who will try to confuse the others. The problem is to find an algorithm to ensure that the loyal generals will reach agreement. It is shown that, using only oral messages, this problem is solvable if and only if more than two-thirds of the generals are loyal; so a single traitor can confound two loyal generals. With unforgeable written messages, the problem is solvable for any number of generals and possible traitors. Applications of the solutions to reliable computer systems are then discussed.

ACM Transactions on Programming Languages and Systems, Vol. 4, No. 3, July 1982, Pages 382-401.

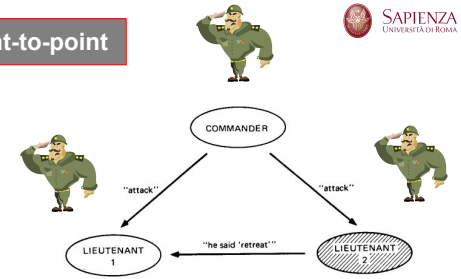


Fig. 1. Lieutenant 2 a traitor.

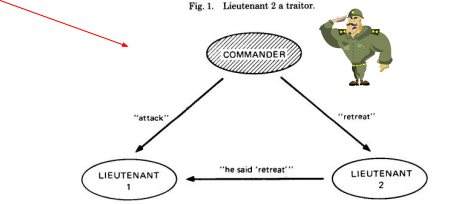


Fig. 2. The commander a traitor.

Byzantine Generals Problem. A commanding general must send an order to his $n - 1$ lieutenant generals such that

IC1. All loyal lieutenants obey the same order.
IC2. If the commanding general is loyal, then every loyal lieutenant obeys the order he sends.

Quanti traditori possiamo tollerare?

LEMMA: non esiste soluzione con 3m+1 generali con > m traditori

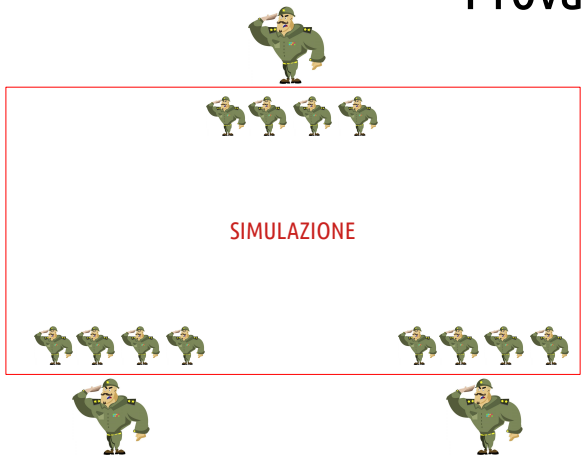
Prova:

- Assumiamo che una soluzione esista
- Usiamo la soluzione per il caso 1 traditore 3 generali

CONTRADDIZIONE

https://youtu.be/_e4wNoTV3Gw

Prova

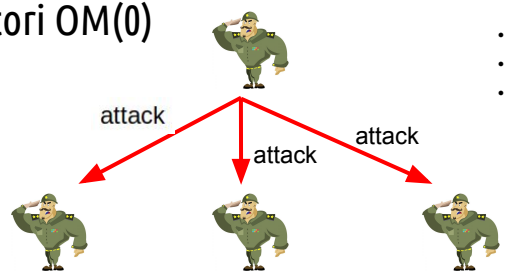


- Assumiamo soluzione per 4 traditori (m) e 12 generali (3m+1)
- Ciascun generale simula 4 generali
- Applichiamo la soluzione sui generali simulati
- Ciascun generale sceglie la soluzione dei suoi generali simulati

Soluzione (per induzione)

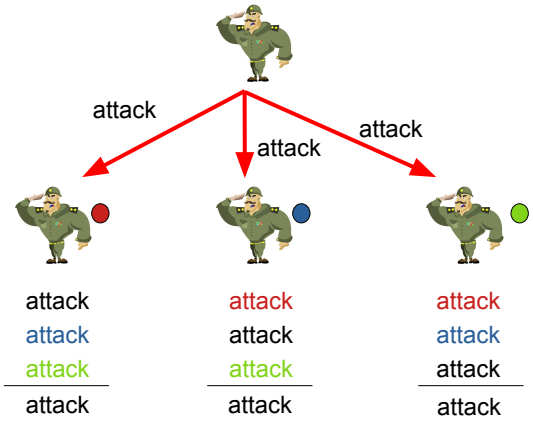
A1. Every message that is sent is delivered correctly.
A2. The receiver of a message knows who sent it.
A3. The absence of a message can be detected.

- Oral Messages OM(m): soluzione per BGP con <=m traditori
- No traditori OM(0)



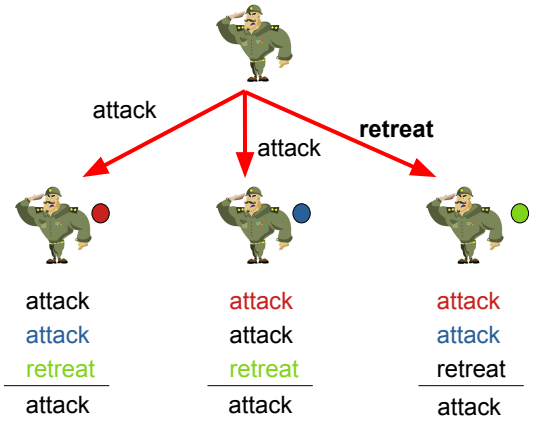
- OM(0)
- C invia ordine
- L eseguono

Soluzione (per induzione)



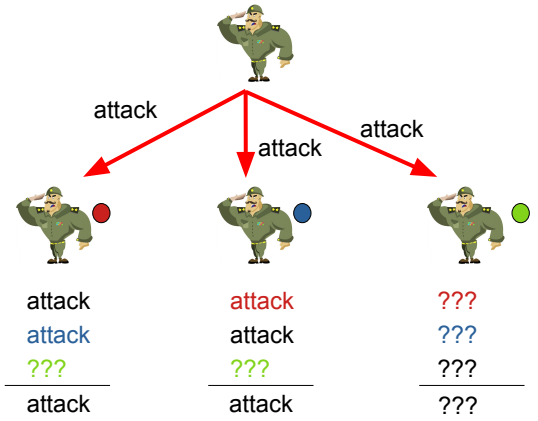
- OM(m), m>0
- C invia ordine
- L
  - Registra ordine
  - Usa OM(m-1) per informare gli altri
  - Usa maggioranza

Soluzione (per induzione)



- OM(m), m>0
- C invia ordine
- L
  - Registra ordine
  - Usa OM(m-1) per informare gli altri
  - Usa maggioranza

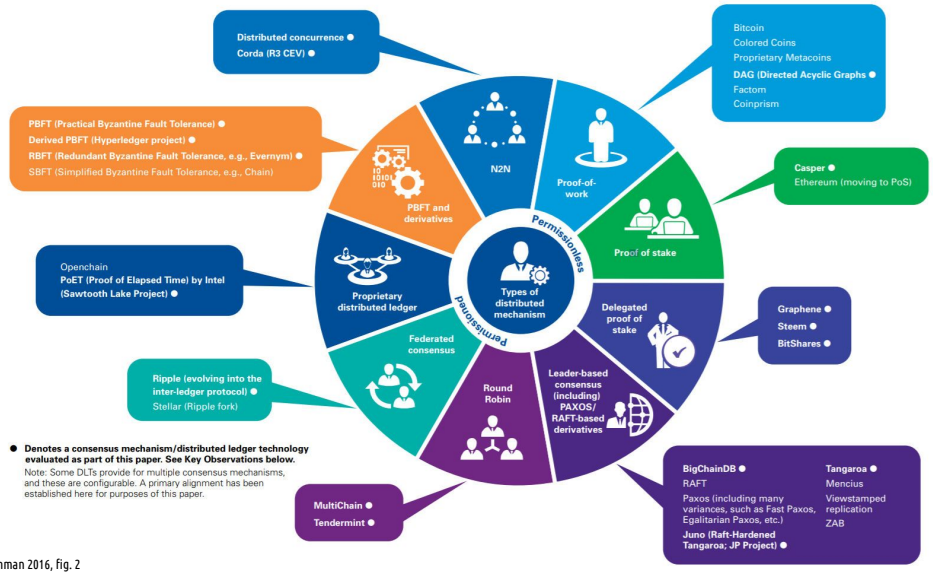
Soluzione (per induzione)



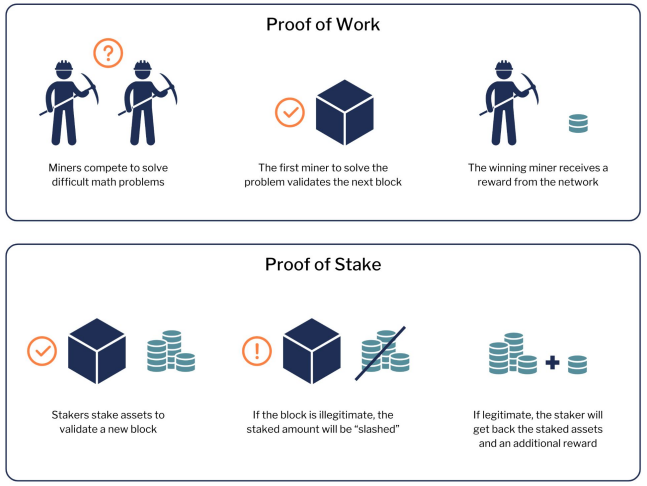
- OM(m), m>0
- C invia ordine
- L
  - Registra ordine
  - Usa OM(m-1) per informare gli altri
  - Usa maggioranza

Complessità (# di messaggi)

m (traditori)	# messaggi
0	n (uno per ogni comandante)
1	n^2
2	n^3
3	n^4
...	...



Source: Seibold and Samman 2016, fig. 2



Delegated Proof-of-Stake

The community empowers a few special users, the *delegates*, to choose the next block, at least for a while.

Hopefully, the chosen delegates are honest to begin with. However, relying on delegates remaining honest for a long time is risky.

Even assuming that there is an ironclad guarantee that all the delegates will remain honest forever, they can easily be attacked. In particular, they can be brought down by a denial of service (DoS) attack.

https://www.algorand.com/Core%20Tech%20in%20a%20Nutshell_2.pdf

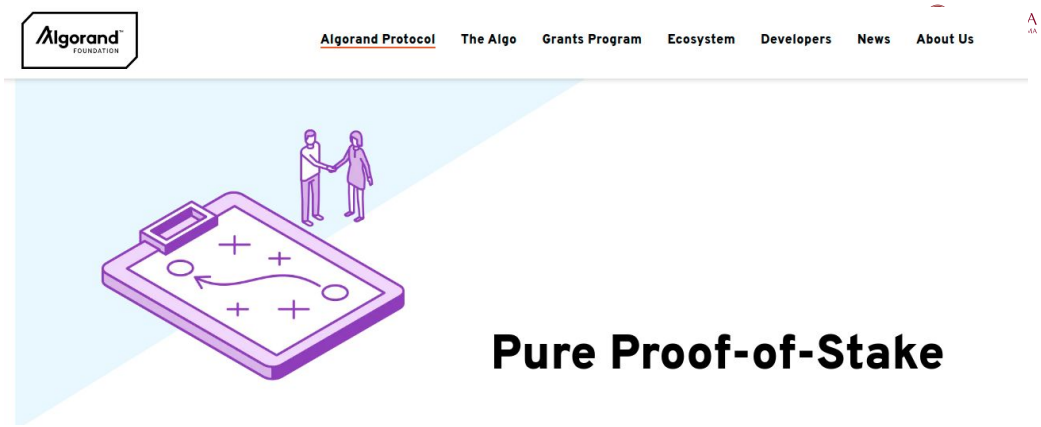
Bonded Proof-of-Stake

Bonded PoS allows 20 users, 200 users, as many as are willing, to put some money on the table — a bond — where they can no longer touch it. These are the users who select the next block on behalf of all of us. If they misbehave, their money is confiscated.

How much would you put hostage on the table → probably relatively small

Big thieves with deep pockets can put a disproportionate amount of money on the table for the sole purpose of controlling the blockchain. They can possibly lose everything → high risk, high profit?

https://www.algorand.com/Core%20Tech%20in%20a%20Nutshell_2.pdf



Pure PoS: Algorand approach

Pure PoS does not try to keep users honest by the fear of imposing fines. Rather, it makes cheating by a minority of the money impossible and cheating by a majority of the money stupid.

In Algorand, only the owners of the majority of the money could prevent other users from transacting. But if they did so, the reputation of the currency would be greatly harmed, the currency would no longer universally accepted, and its purchasing power would be greatly diminished. Not a good outcome for the owners of the majority of the money.

https://www.algorand.com/Core%20Tech%20in%20a%20Nutshell_2.pdf

Algorand pure-proof-of-stake

At a very high level, in Algorand, a new block is constructed in two phases.

- In the first phase, a single token (i.e. Algo) is randomly selected, and its owner is the user who proposes the next block.
- In the second phase, 1000 tokens are selected among all tokens (i.e. Algos) currently in the system. The owners of these 1000 tokens are selected to be part of a phase-2 'committee,' which approves the block proposed by the first user.

https://www.algorand.com/Core%20Tech%20in%20a%20Nutshell_2.pdf

Algorand

← Algorand Dev Portal

Algorand Developer Docs
Start Building
Explore Features
Run a Node
Reference Docs
Algorand Consensus
Community Projects

Algorand Consensus

The Algorand blockchain uses a decentralized Byzantine Agreement protocol that leverages pure proof of stake (Pure POS). This means that it can tolerate malicious users, achieving consensus without a central authority, as long as a supermajority of the stake is in non-malicious hands. This protocol is very fast and requires minimal computational power per node, giving it the ability to finalize transactions efficiently.

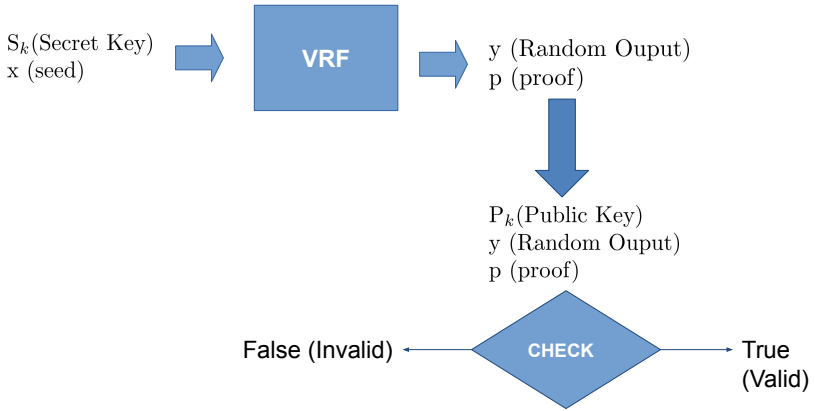
Before getting into detail on the protocol, we discuss two functional concepts that Algorand uses. This is a simplified version of the protocol that covers the ideal conditions. For all technical details see the white paper or the source code.

Verifiable Random Function

Recently we released the source code for our implementation of a Verifiable Random Function (VRF). The VRF takes a secret key and a value and produces a pseudorandom output, with a proof that anyone can use to verify the result. The VRF functions similar to a lottery and is used to choose leaders to propose a block and committee members to vote on a block. This VRF output, when executed for an account, is used to sample from a binomial distribution to emulate a call for every algo in a user's account. The more algos in an account, the greater chance the account has of being selected--it's as if every algo in an account participates in its own lottery. This method ensures that a user does not gain any advantage by creating multiple accounts.

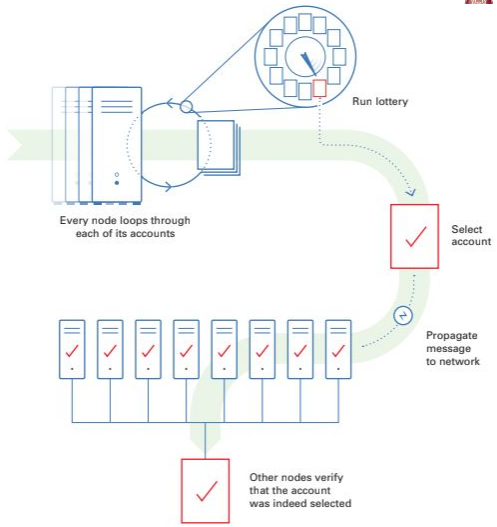
https://developer.algorand.org/docs/algorand_consensus/

Verifiable Random Function (VRF)



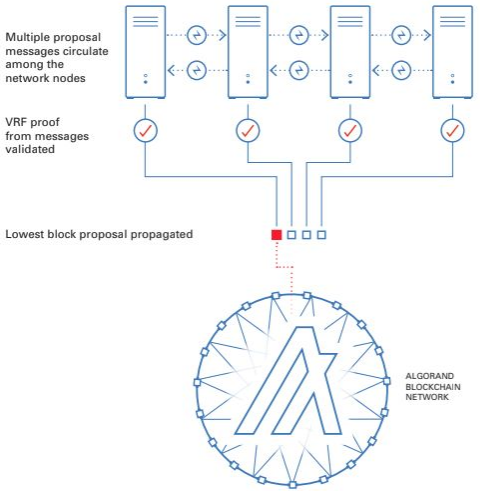
Block Proposal

- Propose of new blocks by selected (VRF) accounts
- The VRF acts similar to a weighted lottery where **the number of Algos that the account has participating online determines the account's chance of being selected**.
- Once an account is selected by the VRF, the node propagates the proposed block along with the VRF output, which proves that the account is a valid proposer.



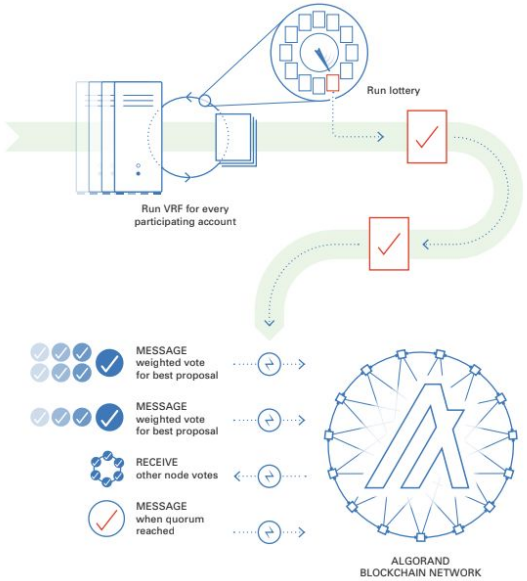
Soft Vote: Select one proposal

- Each node in the network will get many proposal messages from other nodes.
- Nodes will verify the signature of the message and then validate the selection using the VRF proof.
- Next, the node will compare the hash from each validated winner's VRF proof to determine which is the lowest and will only propagate the block proposal with the lowest VRF hash.
- This process continues for a fixed amount of time to allow votes to be propagated across the network



Soft Vote: Select one proposal

- Each node will then run the VRF for every participating account it manages to see if they have been chosen to participate in the soft vote committee.
- If any account is chosen it will have a weighted vote based on the number of Algos the account has, and these votes will be propagated to the network.



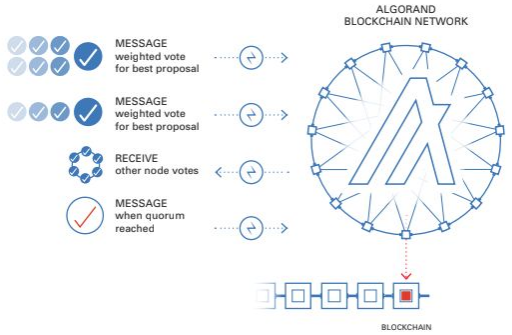
Soft Vote: block approval

Difficult coalitions

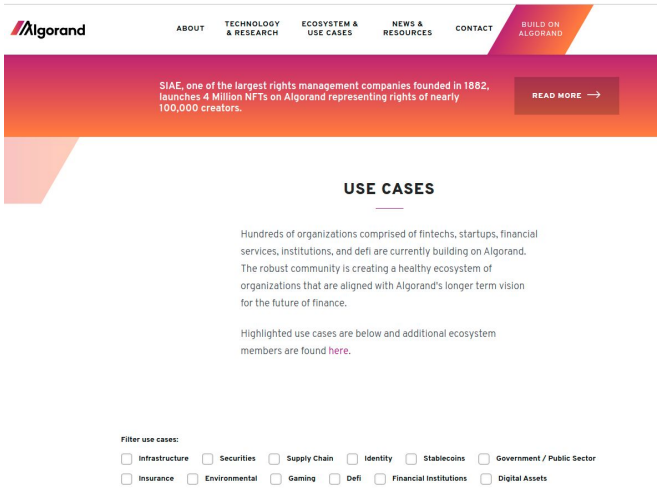
- A **new** committee is selected for every step in the process and each step has a different committee size.
- This committee size is quantified in Algos. A quorum of votes is needed to move to the next step and must be a certain percentage of the expected committee size. These votes will be received from other nodes on the network and each node will validate the committee membership VRF proof before adding to the vote tally. Once a quorum is reached for the soft vote the process moves to the certify vote step.

Certify the vote

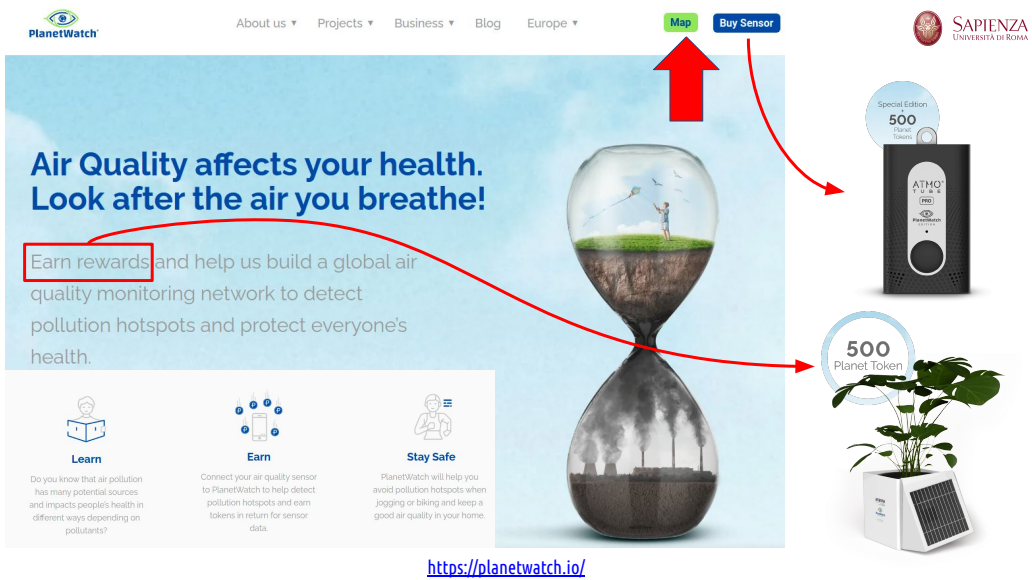
- A new committee checks the block proposal that was voted on in the soft vote stage for overspending, double-spending, or any other problems.
- If valid, the new committee votes again to certify the block. This is done in a similar manner as the soft vote where each node iterates through its managed accounts to select a committee and to send votes.
- These votes are collected and validated by each node until a quorum is reached, triggering an end to the round and prompting the node to create a certificate for the block and write it to the ledger.
- At that point, a new round is initiated and the process starts over



Use-cases



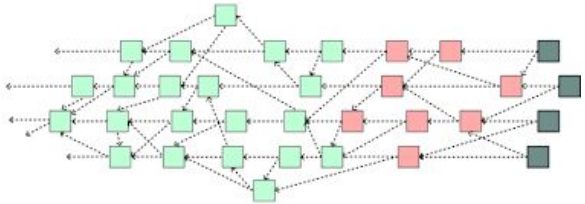
https://www.algorand.com/ecosystem/use-cases



https://planetwatch.io/

IOTA

- The tangle a DAG
- The benefit is twofold: It allows for greater scalability and it eliminates the need to pay transaction fees to miners.



The Tangle: an Illustrated Introduction

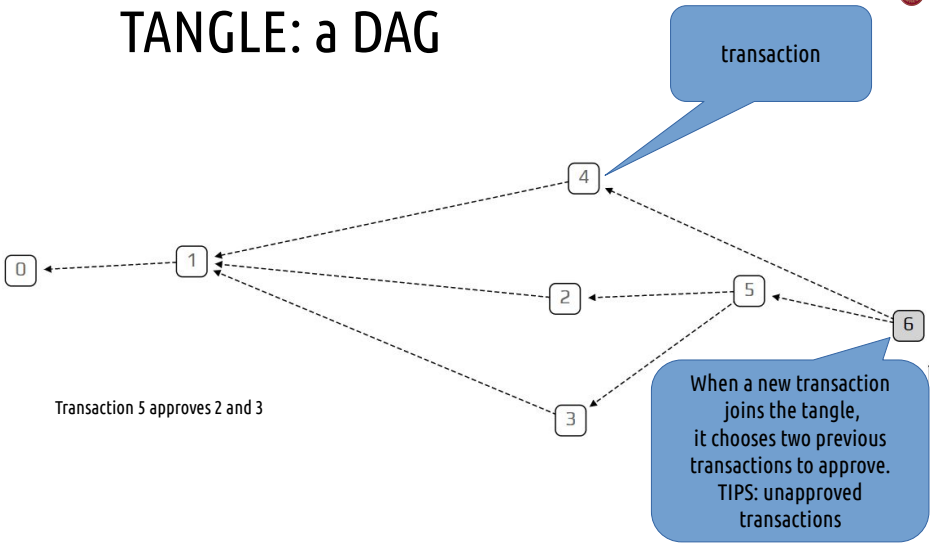
Research & Development Jan 31, 2018

This is the first in a series of beginner level posts, aimed at those who want to learn more about how IOTA works under the hood. We will loosely follow the **whitepaper**, but go a bit slower and add some pictures to clarify the basic concepts. In this article, we introduce the *Tangle*, explain what it is, and how we study it mathematically in the IOTA research team.

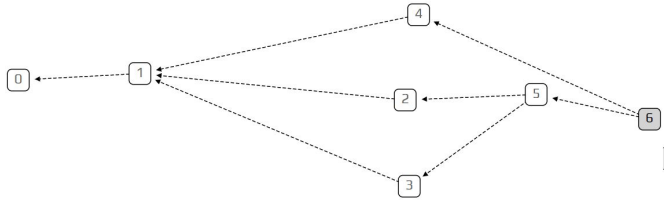
To understand the tangle, we need to learn about what computer scientists call a ***directed graph***. A directed graph is a collection of *vertices* (squares), which are connected to each other by *edges* (arrows). This is an example of a directed graph:

https://blog.iota.org/the-tangle-an-illustrated-introduction-4d5eae6fe8d4/

TANGLE: a DAG



TANGLE: a DAG



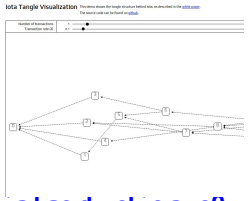
Tips: unapproved transactions (6)

"The strategy for choosing which two tips to approve is very important, and is the key to IOTA's unique technology"

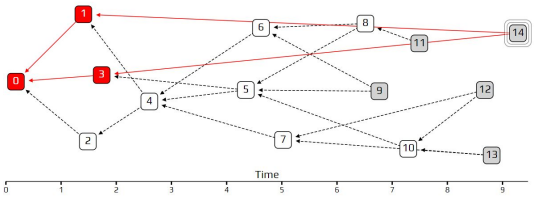
e.g random selection

Simple example: Random choice of tips

- Each incoming transaction looks at all the currently unapproved transactions (i.e. tips), and simply chooses two at random
- Fantastic visual tool https://public-rdsdavdrpd.now.sh/
- Let's move on to the very nice blog at https://blog.iota.org/the-tangle-an-illustrated-introduction-c0a86f994445



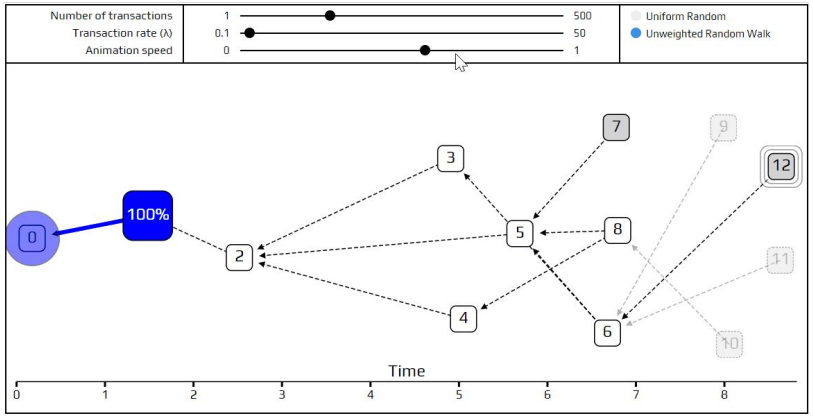
Avoid lazy tips



A lazy tip is one that approves old transactions rather than recent ones → This does not help the network, since no new transactions are confirmed.

- Random selection does not help
- Force participants to only approve recent transactions → clash with the idea of decentralization.

More advanced tip selection: Unweighted Random Walk



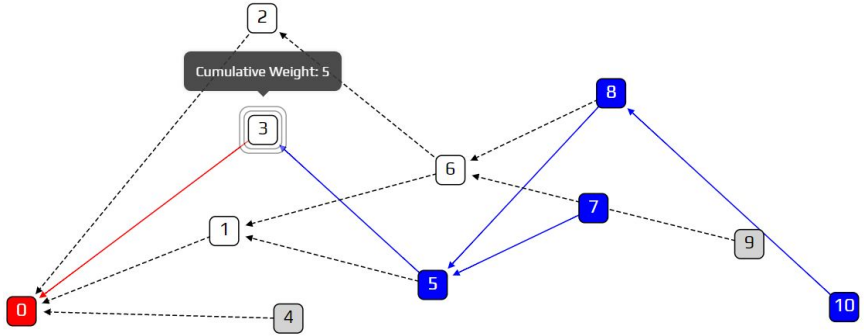
Random walk. Using this algorithm, we put a walker on the genesis transaction, and have it start "walking" towards the tips (i.e select). Very recent transactions, which are "invisible" to the current random walk, are shown as transparent.

Bias our random walk, so we are less likely to choose lazy tips

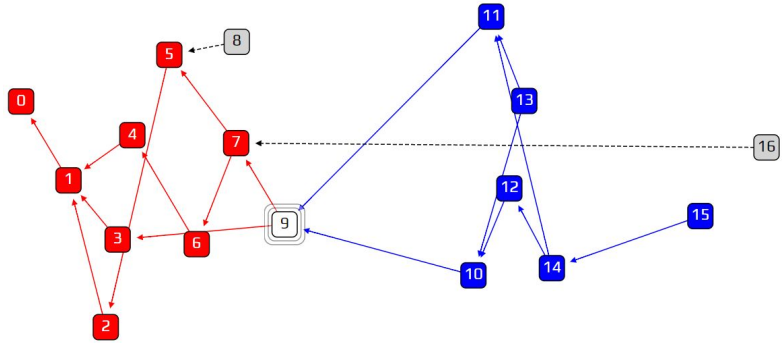
We will use the term *cumulative weight* to denote how important a transaction is.

We are more likely to walk towards a heavy transaction than a light one.

The definition of cumulative weight is very simple: we count how many approvers a transaction has, and add one. We count both direct and indirect approvers. In the example below, transaction 3 has a cumulative weight of five, because it has four transactions which approve it (5 directly; 7, 8, and 10 indirectly).

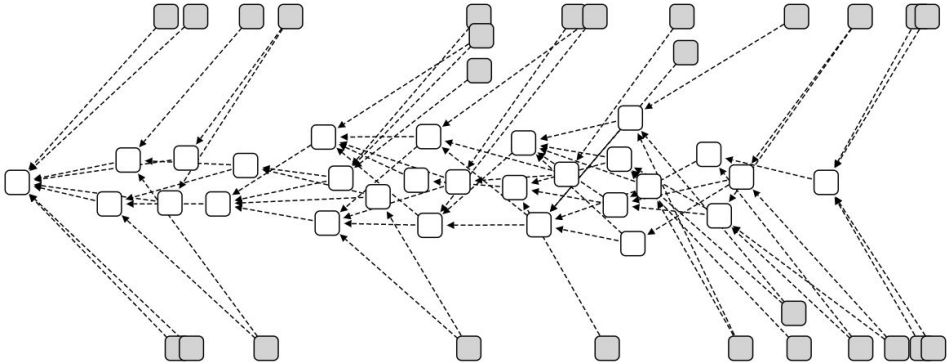


Why it works?



- In order for 16 to get approved, the random walker must reach transaction 7, and then choose transaction 16 over transaction 9.
- But this is unlikely to happen, because transaction 16 has a cumulative weight of 1, and transaction 9 has a cumulative weight of 7! This is an effective way of discouraging lazy behavior.

Risk



The gray squares are tips, with zero approvers. While it is normal to have some tips on the right end of the diagram, it is a problem to see so many of them spread out across the timeline. These tips are transactions that are *left behind*, and will never be approved. This is the down-side to biasing our walk too much: if we insist on choosing only the heaviest transaction at any given point, a large percentage of the tips will never get approved. We are left with only a central corridor of approved transactions, and forgotten tips on the sidelines.

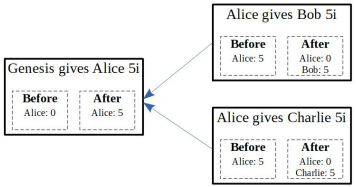
A bias α

We need a method to define how likely we are to walk towards any particular approver at a given junction. The exact formula we choose is not important, as long as we give some bias to heavier transactions, and have a parameter to control how strong this bias is. This introduces our new parameter *α*, which sets how important a transaction's cumulative weight is.

- *α=0* → unweighted walk.
- *α* very high → super-weighted walk.
- In between, we can find a good balance between punishing lazy behavior and not leaving too many tips behind.

Determining an ideal value for *α* is an important research topic in IOTA.

Double spending

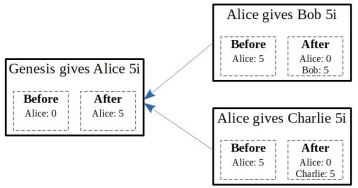


Alice creates two *branches* in the tangle that cannot be reconciled.

This creates a problem for honest users of IOTA: which branch should they approve?

The solution to this problem is once again the weighted walk we learned about last week. **Eventually one of the branches will grow heavier than the other, and the lighter one will be abandoned.** This also implies that a transaction cannot be considered to be confirmed immediately after it is issued, even if it has some approvers, since it might be part of a branch that will be abandoned eventually. In order to be sure your transaction is confirmed, you have to wait for its *confirmation confidence* to be high enough.

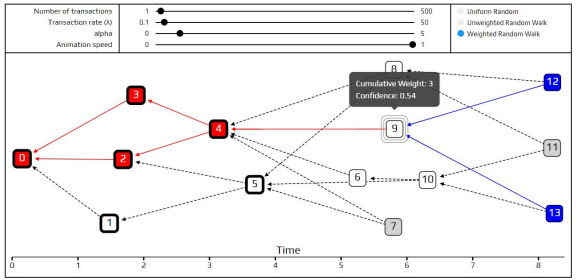
How do Bob and Charlie know if they really got the money from Alice?

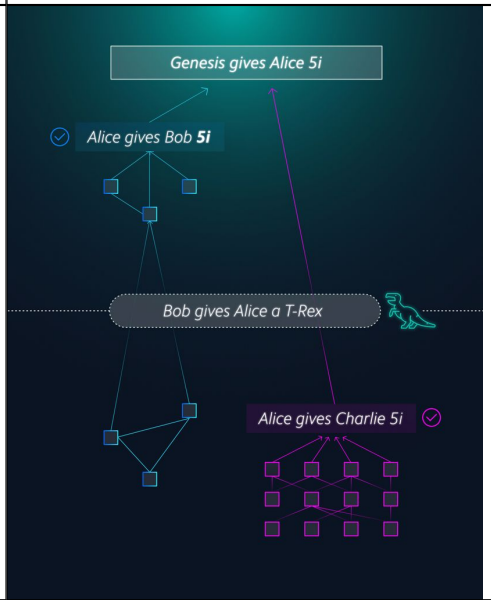


Eventually one of the branches will grow heavier than the other, and the lighter one will be abandoned.

Confirmation confidence

1. Run the tip selection algorithm 100 times.
2. Count how many of those 100 tips approve our transaction, and call it A.
3. The confirmation confidence of our transaction is "A percent".

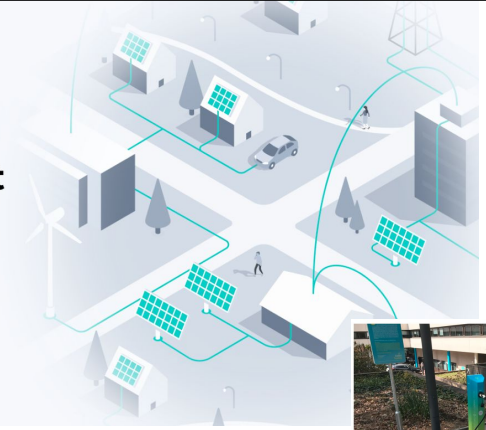




IOTA

IOTA-powered Smart Grid Infrastructure

Enabling Energy Trading Platforms



https://blog.iota.org/iota-powered-smart-grid-infrastructure/

The mini Tesla connected to the IOTA Charger

https://blog.iota.org/worlds-first-iota-smart-charging-station-52f9024db788/

Iota use cases

- Bosch — the Bosch XDK (Cross Domain Development Kit) is a programmable sensor device and IoT prototyping platform used to collect specific, real-time data which can then be sold via the IOTA Data Marketplace.
- Fujitsu — the company is using the IOTA protocol in a proof-of-concept, immutable data storage medium for audit trails across industrial production environments and supply chains.

https://coindelegraph.com/news/how-significant-is-blockchain-in-internet-of-things