Distributed Ledgers

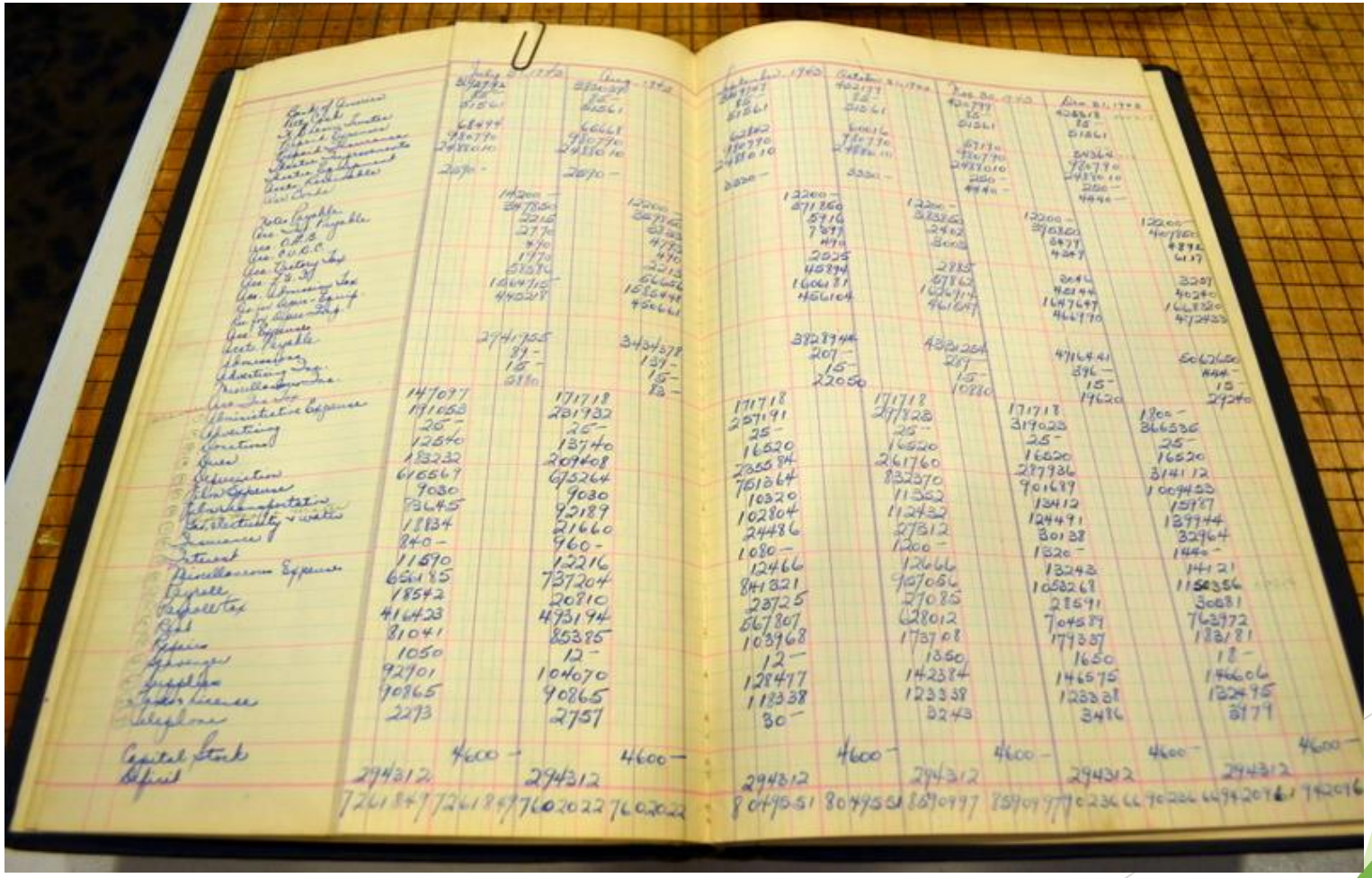
Dr. E. Markakis

Content

- Distributed Ledger Fundamentals and HyperLedger Introduction
  - Presented BY: Manjunath N V yoda@security-exploits.com
- HyperLedger Setup tools
- H2020 CHARIOT (IoT Distributed Ledger)
  - Reference IoT Scenario the Chariot H2020 Use case
- What a Survey paper is
  - A Survey on Distributed ledger technologies for Internet of Things IoT
- Reading Material For Blockchains

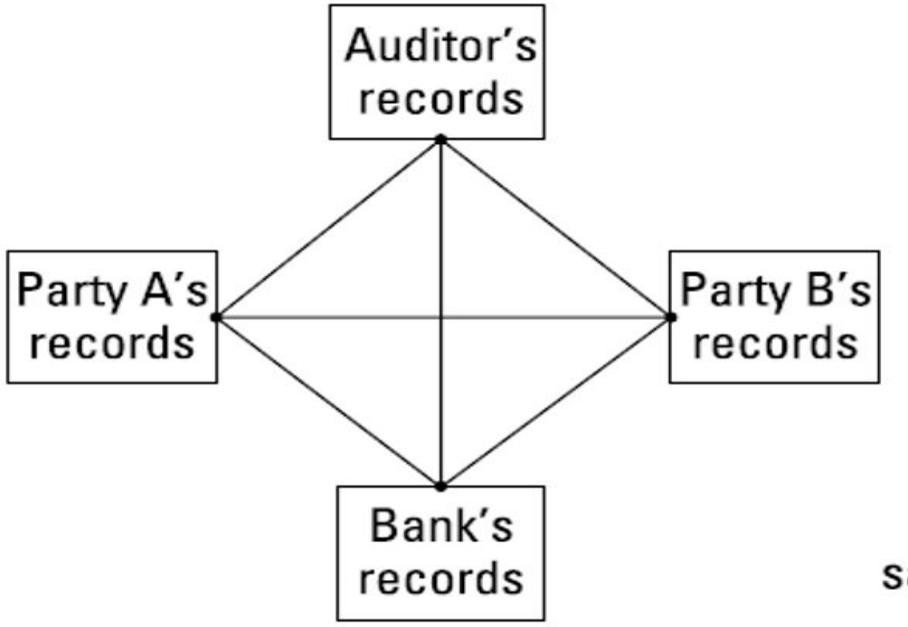
Distributed Ledger Introduction

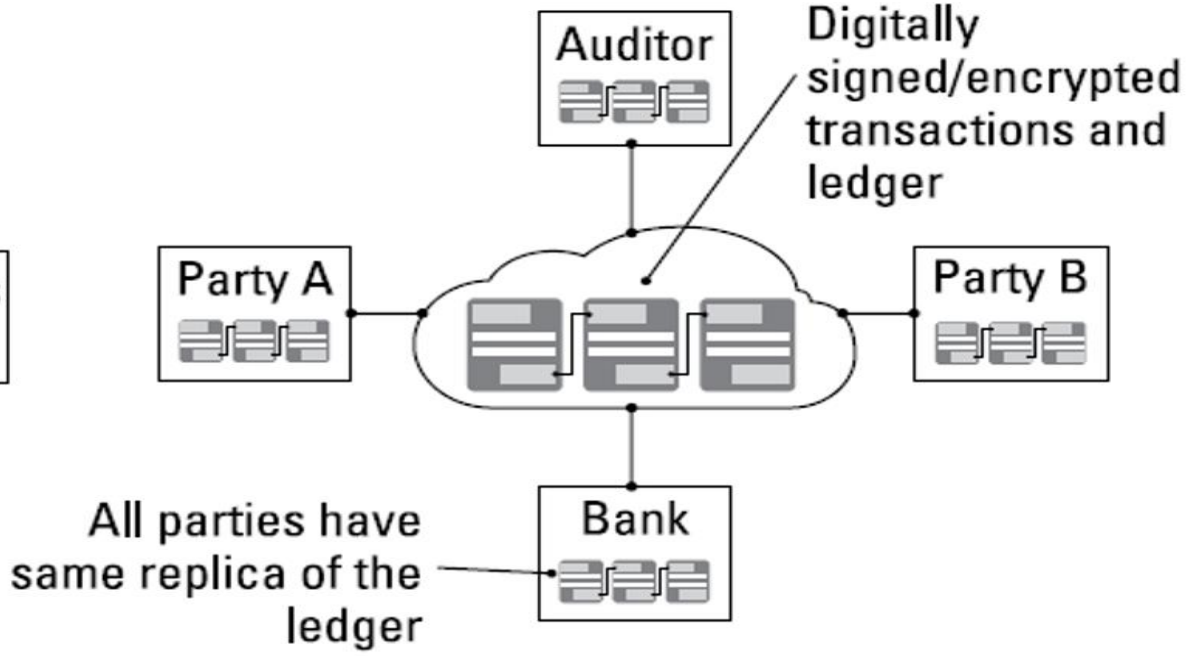
Ledger



Tracing Blockchain's Origin

- The shortcomings of current transaction systems
  - During 2000's financial crisis





Bitcoin Whitepaper: 10/31/2008

https://bitcoin.org/bitcoin.pdf

Bitcoin: A Peer-to-Peer Electronic Cash System

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

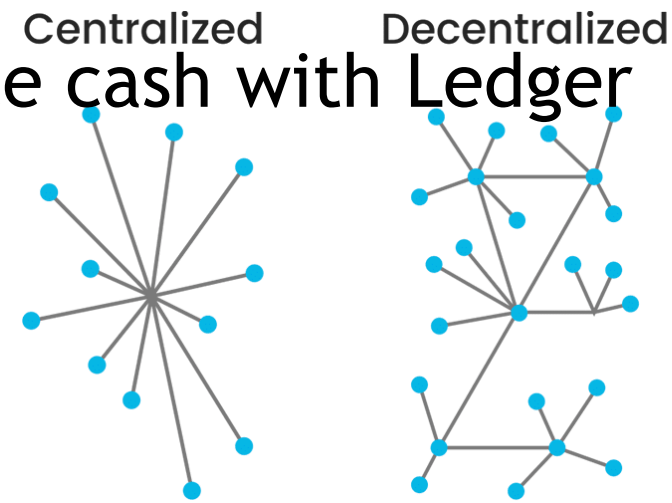
Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required 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.

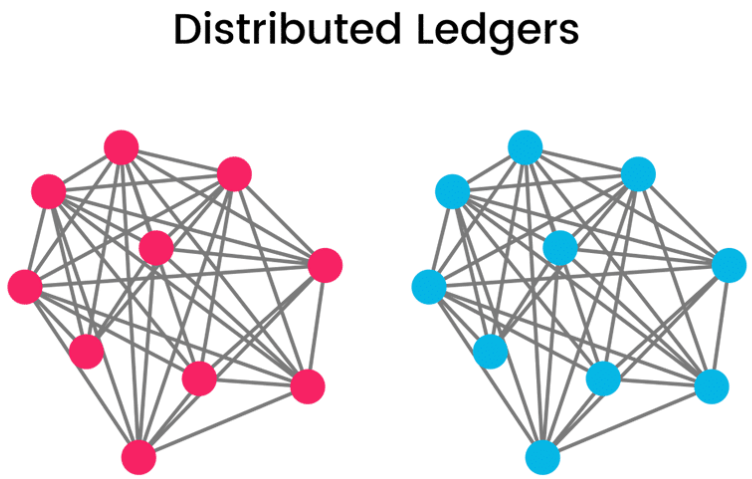
The Long Road to Bitcoin

- Centralized Banking: not robust
- Satoshi determined to find the centralized part of banks
  - The ledger
  - “What if I could turn a bank inside out? Instead of one central party controlling the ledger, what if every user were recruited to maintain a constantly updated copy?”
- The strength of the digital was perfect copies, so copy the ledger, everywhere, instantly.
  - Any ledgers with even one common not agreeing with the masses would be discarded, leaving fraudsters powerless
- **Replace cash with Ledger!**

Decentralization

- Replace cash with Ledger





The New Networks

Distributed ledgers can be public or private and vary in their structure and size.

Public blockchains

Require computer processing power to confirm transactions ("mining")

- Users (●) are anonymous

- Each user has a copy of the legder and partipates in confirming transactions independently

- Users (●) are not anonymous

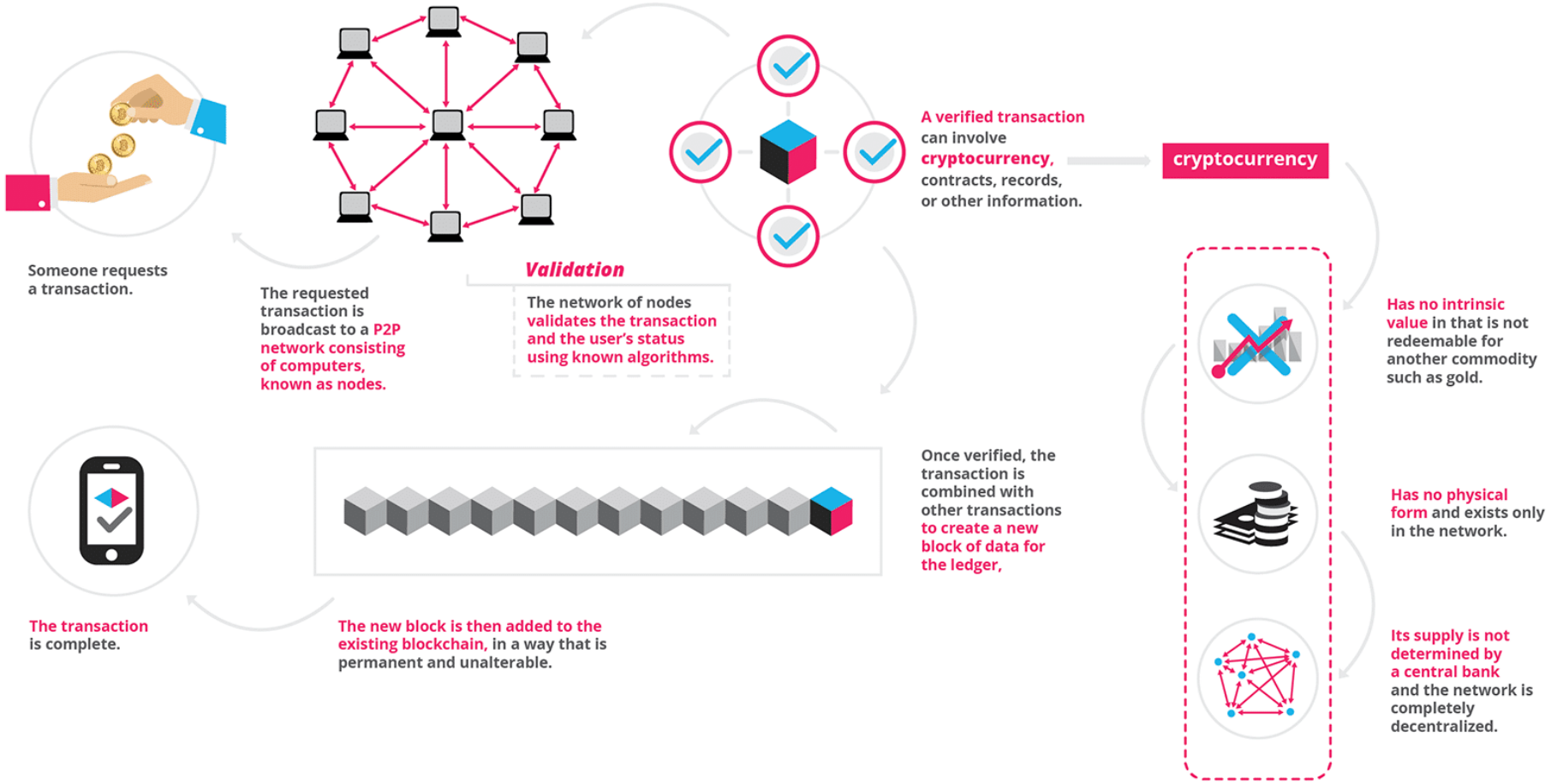
- Permision is required for users to have a copy of the legder and participate in confirming transactions

Blockgeeks

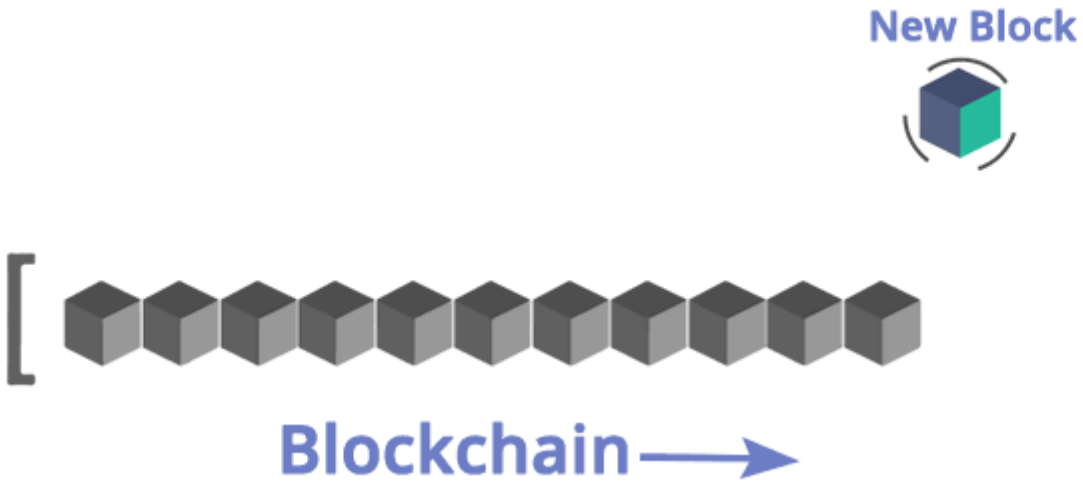
The decentralized ledger

- Decentralization: get rid of the Third Party
- Satoshi paired two main technologies
  - Proof of Work: to solve the double spending problem
  - Elliptic Curves: to solve unique access to the ledger
- Nothing was newer than 2001
  1. 2001: SHA-256 finalized
  2. 1999-present: Byzantine fault tolerance
  3. 1999-present: P2P networks
  4. 1998: Wei Dai, B-money
  5. 1998: Nick Szabo, Bit Gold
  6. 1997: HashCash
  7. 1992-1993: Proof-of-work for spam
  8. 1991: cryptographic timestamp
  9. 1980: public key crypto algorithm

Highlights

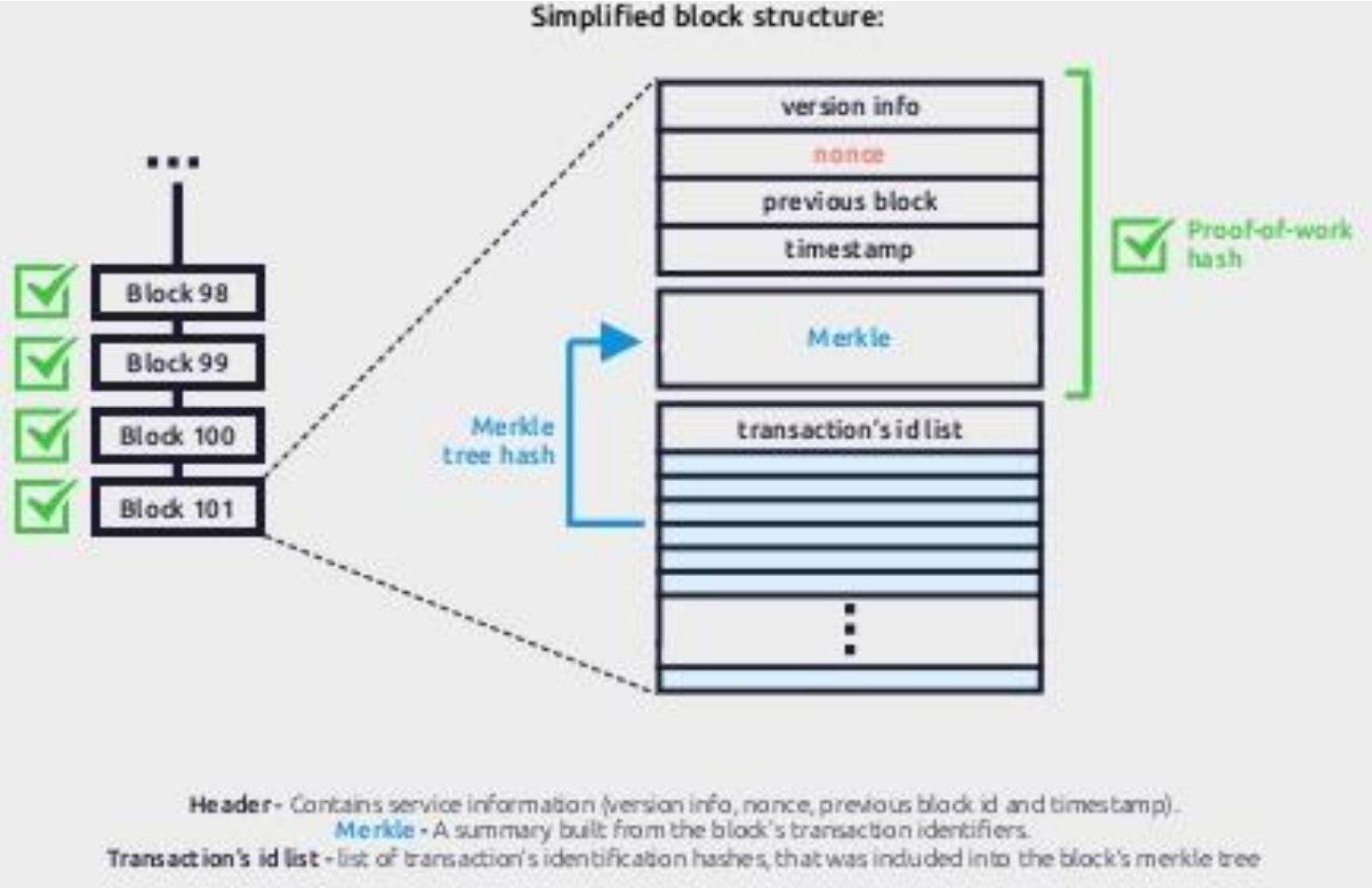

Someone requests a transaction.
The requested transaction is broadcast to a P2P network consisting of computers, known as nodes.
Validation
The network of nodes validates the transaction and the user's status using known algorithms.
A verified transaction can involve cryptocurrency, contracts, records, or other information.
cryptocurrency
Has no intrinsic value in that is not redeemable for another commodity such as gold.
Has no physical form and exists only in the network.
Its supply is not determined by a central bank and the network is completely decentralized.
Once verified, the transaction is combined with other transactions to create a new block of data for the ledger,
The new block is then added to the existing blockchain, in a way that is permanent and unalterable.
The transaction is complete.

Blockchain Structure

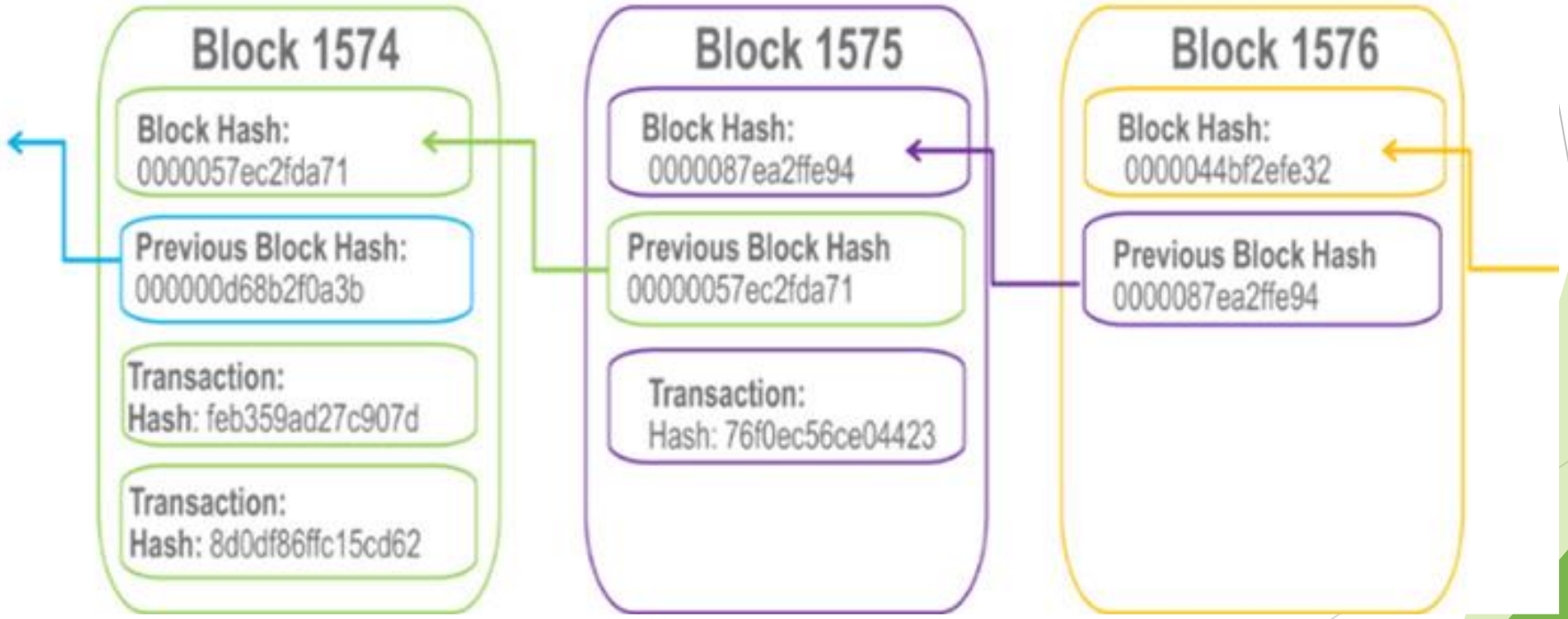


Source: https://www.edureka.co/blog/blockchain-tutorial/

What does a block look like?



Why It's Called "Blockchain"



Blockchain Architecture

- Revolutionary Technology
  - Protocol
  - TCP/IP, HTTP, Cloud Computation, Big Data, IoT, FinTech…
- Melanie Swan: Blockchain: Blueprint for A New Economy, Jan 2015
  - Blockchain 1.0
    - Bitcoin
    - Programmable Money
  - Blockchain 2.0
    - Ethereum
    - Smart Contract
  - Blockchain 3.0…
    - Non-Financial Uses
- Applications

Key Concepts of Cryptocurrency

Distributed shared ledger

Cryptography

Consensus

Smart contracts

Source: IBM, A new disruption in financial services

Building trust with blockchain

- Distributed and sustainable:
- Secure, private, and indelible:
- Transparent and auditable:
- Consensus-based and transactional:
- Orchestrated and flexible:

Different Players in Implementation

- Blockchain user
- Regulator
- Blockchain developer
- Blockchain network operator
- Traditional processing platforms
- Traditional data sources
- Certificate authority

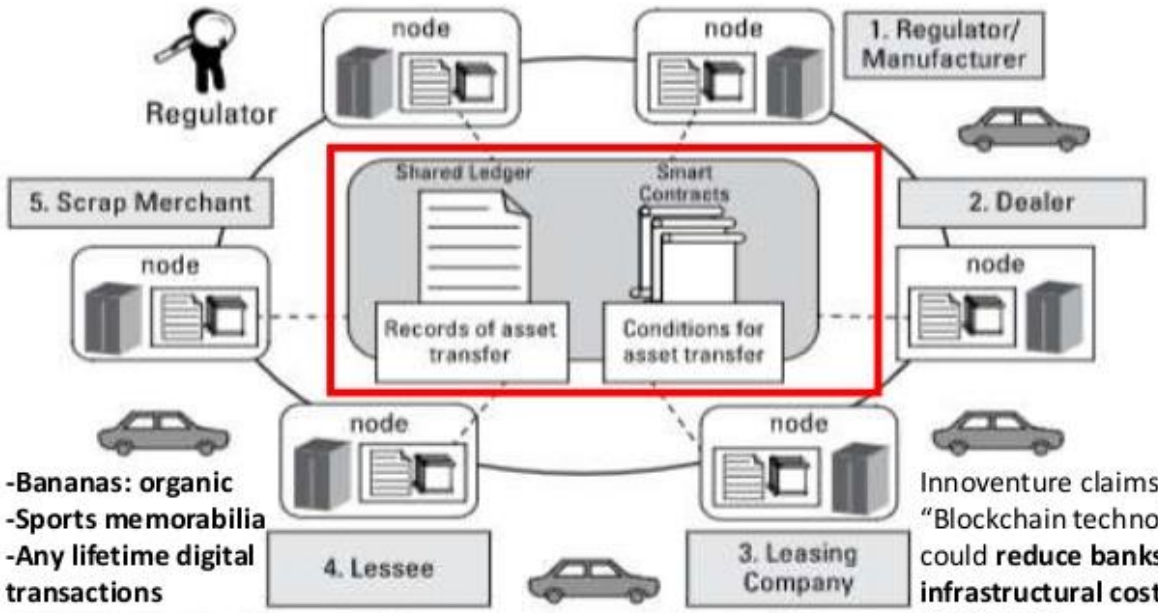


FIGURE 1-3: Tracking vehicle ownership with blockchain.

-Bananas: organic
-Sports memorabilia
-Any lifetime digital transactions

Innoventure claims that "Blockchain technology could **reduce banks' infrastructural costs by $1.520 trillion a year by 2022**" [2]

Block Chain usecase (dubai)

HyperLedger Introduction

Hyperledger Fabric

- The Linux Foundation founded Hyperledger in 2015
- Hyperledger Fabric is a platform for distributed ledger solutions in industrial level.
- A modular architecture - Delivers high degrees of confidentiality, resiliency, flexibility and scalability.
- It is designed to support pluggable implementations of different components, and accommodate the complexity and intricacies that exist across the economic ecosystem.
- Breaks from some other blockchain systems is that it is **private** and **permissioned**

Hyperledger Fabric - Cont.

- Like other blockchain technologies, it has a ledger, uses smart contracts, and is a system by which participants manage their transactions.
- Ledger data can be stored in multiple formats, consensus mechanisms can be switched in and out.
- Offers the ability to create **channels**, allowing a group of participants to create a separate ledger of transactions.
- Hyperledger is based on blockchain but its not a crypto currency.
- There is no mining, just order system do it.
- Operational power: 0.5 million operations per minute where as other blockchain does only 1000.

In Summary

- Hyperledger Fabric is enterprise grade distributed ledger based on blockchain technologies that uses smart contracts to enforce trust between parties.
- Hyperledger in general do not enforce any requirements about the hardware, network infrastructures, additional software around it, security models etc.
- No concept of computational power.

Advantages of Hyperledger Fabric

- Permissioned membership
- Performance, scalability, and levels of trust
- Data on a need-to-know basis
- Rich queries over an immutable distributed ledger
- Modular architecture supporting plug-in components
- Protection of digital keys and sensitive data

Hyperledger Components

- Fabric CA,
- Peer
- Ordering service
- Channel
- Chaincode

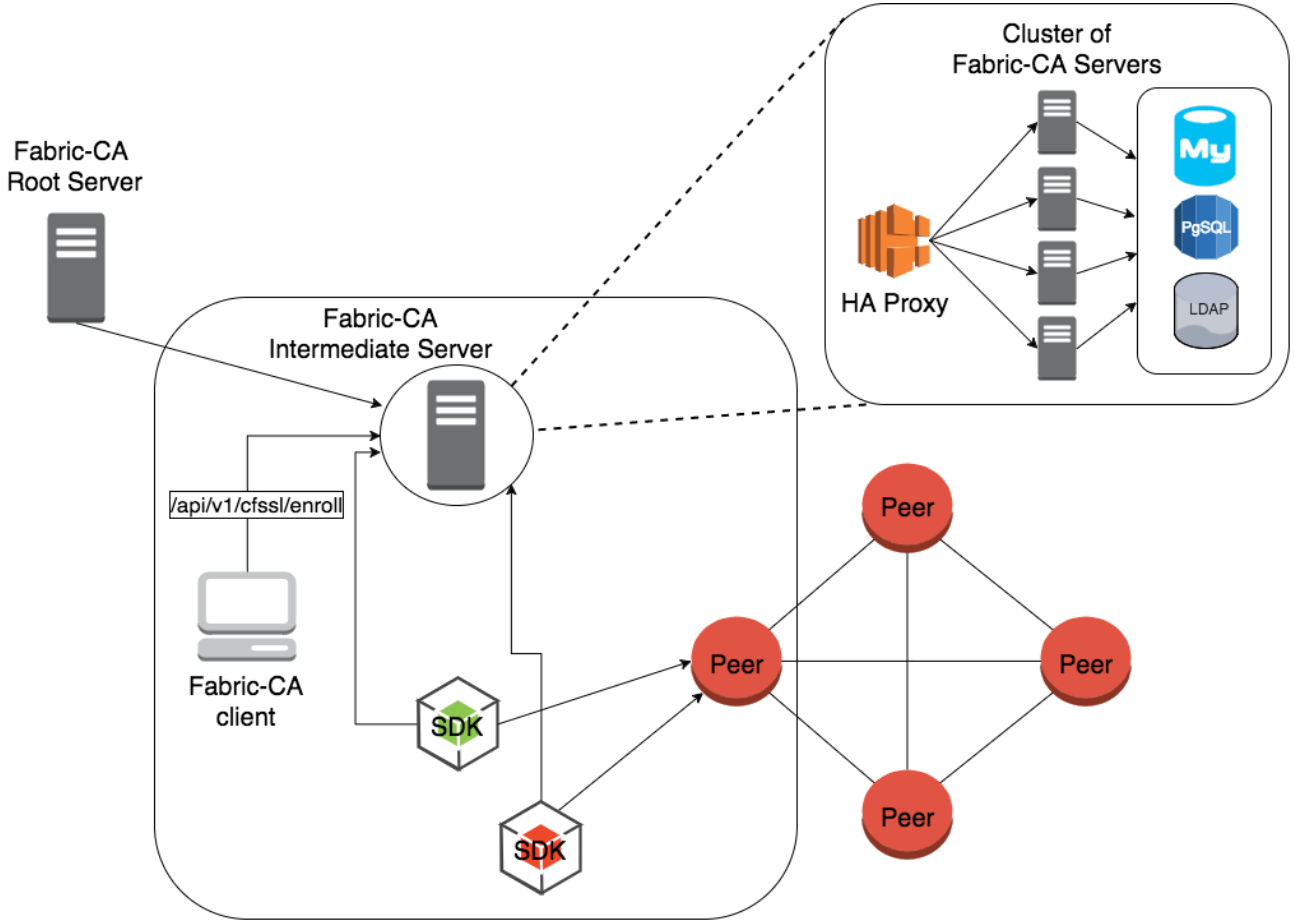
Fabric CA

The Hyperledger Fabric CA is a Certificate Authority (CA) for Hyperledger Fabric.

It provides features such as:

- registration of identities, or connects to LDAP as the user registry
- issuance of Enrollment Certificates (ECerts)
- certificate renewal and revocation
- consists of both a server and a client component.

CA - WorkFlow


Cluster of
Fabric-CA Servers
Fabric-CA
Root Server
HA Proxy
My
PgSQL
LDAP
Fabric-CA
Intermediate Server
/api/v1/cfssl/enroll
Fabric-CA
client
SDK
SDK
Peer
Peer
Peer
Peer

CA cont.

- Every single operation that is executed inside Hyperledger fabric must be cryptographically signed with this certificate.
- You can add attributes, roles
- Certificates are X.509 standards.
- You can remove the necessity of certificates if you don’t need it.
- Chaincodes read this data and make business decisions.

Peer

- Peer is the place where the ledger and the blockchain data is stored.
- You must have more than one peer in production.
- One peer may be part of many channels.
- Every single channel is inside the peer.
- It endorse any update of the ledger.
- You can create backup of the ledger from the peer

Ordering Service

- Ordering service is the heart of consensus algorithm and the heart of hyper ledger fabric.
- Main role is to provide the order of operations.
- before committing anything to ledger it must pass through the ordering service.
- it is responsible for verification, security, policy verification etc.

Channel

- **Channel** is a private “subnet” of communication between two or more specific network members.
- A channel is defined by members (organizations), anchor peers per member, the shared ledger, chaincode application(s) and the ordering service node(s).
- Each peer that joins a channel, has its own identity given by a membership services provider (MSP).

Channel cont.

- channels are completely isolated,
- they have different ledgers, different height of blocks, policies, stories, rules.
- completely isolated instance of hyper ledger fabric.
- never exchange data.
- outside of a channel , one can’t even see that there is a channel.
- you can make a policy who can see the data in the channel and who can make an operation.
- every single party inside a channel must agree about other parties.

Channel configuration properties

- **Versioned**: All elements of the configuration have an associated version which is advanced with every modification. Further, every committed configuration receives a sequence number.
- **Permissioned**: Each element of the configuration has an associated policy which governs whether or not modification to that element is permitted. Anyone with a copy of the previous configtx (and no additional info) may verify the validity of a new config based on these policies.
- **Hierarchical**: A root configuration group contains sub-groups, and each group of the hierarchy has associated values and policies. These policies can take advantage of the hierarchy to derive policies at one level from policies of lower levels.

Chaincode

- A chaincode typically handles business logic agreed to by members of the network, so it similar to a “smart contract”.
- All your business logic is inside the chaincode.
- Its written in Go. Implementation of java and javascript are on the way.
- Chaincode me installed in every peer and channel.
- Policy must be provided.

Hyperledger Composer

- Hyperledger Composer is a set of collaboration tools for building blockchain business networks that make it simple and fast for business owners and developers to create smart contracts and blockchain applications to solve business problems

- Extensive
- Open development toolset and
- Framework to make developing Blockchain applications easier.

Hyperledger setup

Implementation

- https://hyperledger-fabric.readthedocs.io/en/release-1.4/

Installing the pre-requisites

- Install cURLDocker Engine: Version 17.03 or higher
- Docker and Docker Compose: 17.06.2-ce or greater
- Go Programming Language
  - Go version 1.12.x is required.
- Node.js 10.15.3 and higher
- Python: 2.7.x

Reference IoT Scenario the Chariot H2020 Use case

Reference IoT Scenario

Use Case: Manufacturing Plant

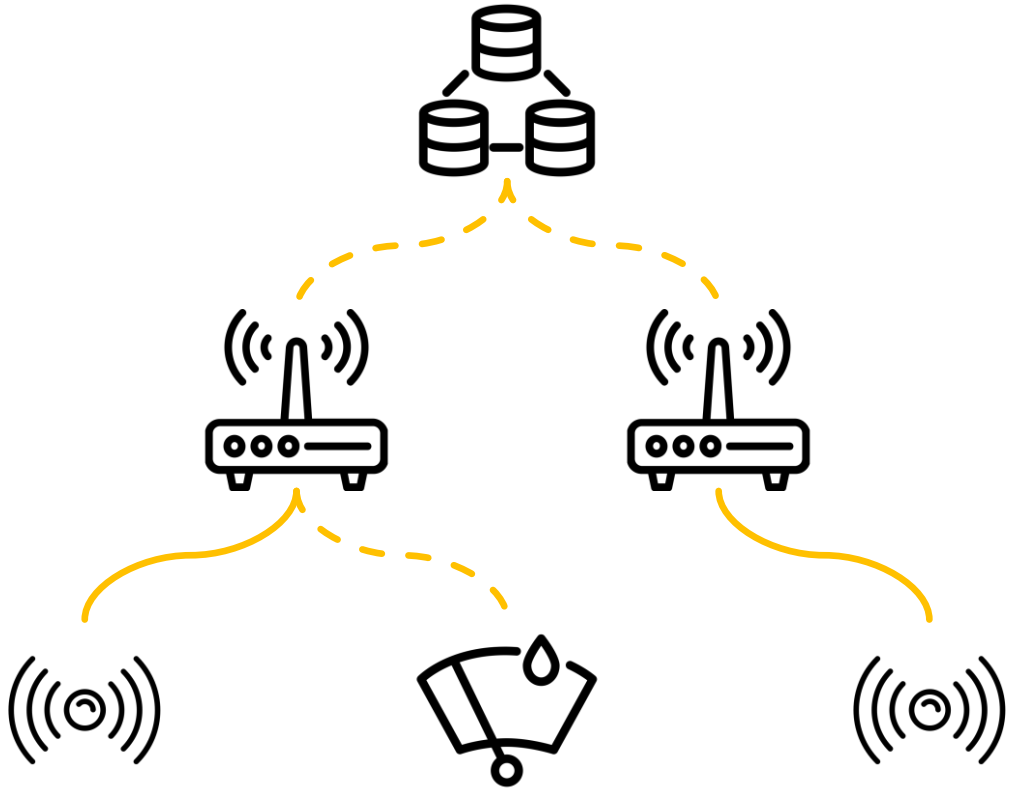
- Industry 4.0 stage with the usage of IoT
- Predictive maintenance by data analytics and machine learning
- Reduced operational cost due to properly allocated resources for maintenance
- Increased revenue due to higher efficacy of manufacturing tools
- Increased customer satisfaction as a result of faster demand fulfillment times
- Reduced risk thanks to stricter operational compliancy
- Correct Operation? Bona Fide Data? Malicious Devices?
- All the above can lead to the exact opposite of the aforementioned benefits

Approach to Securing IoT Network

Legend

Wireless

Wired

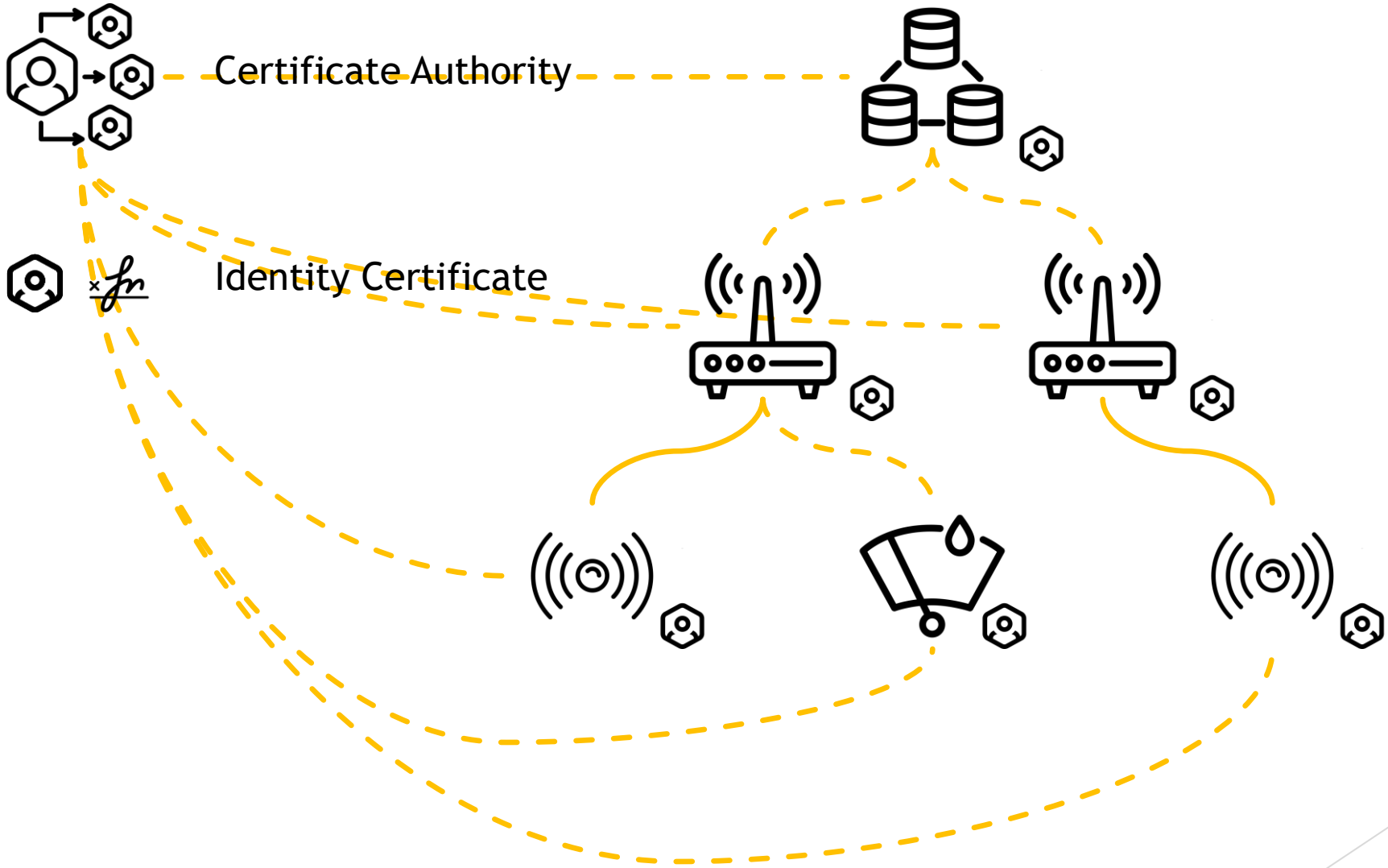


Data Aggregators

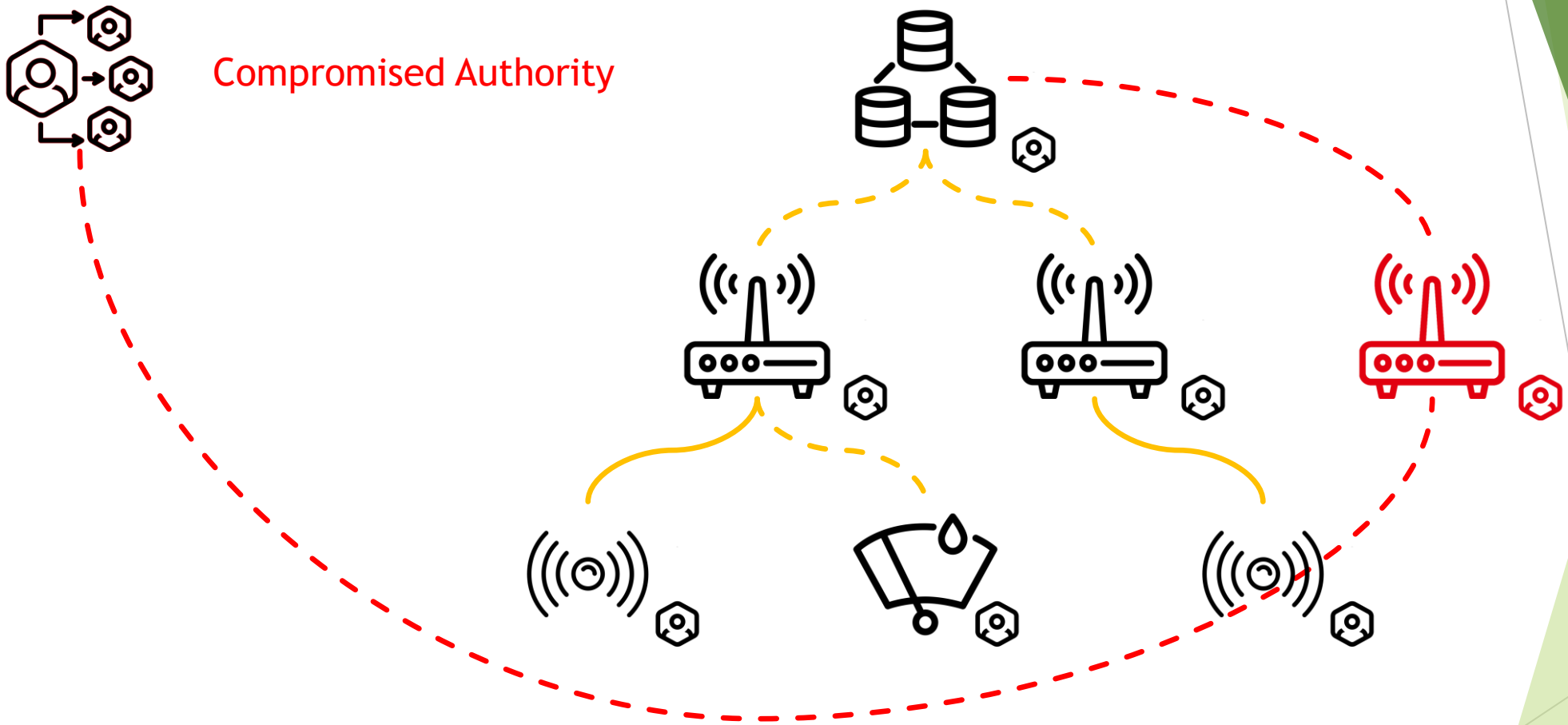
IoT Gateways

IoT Sensors

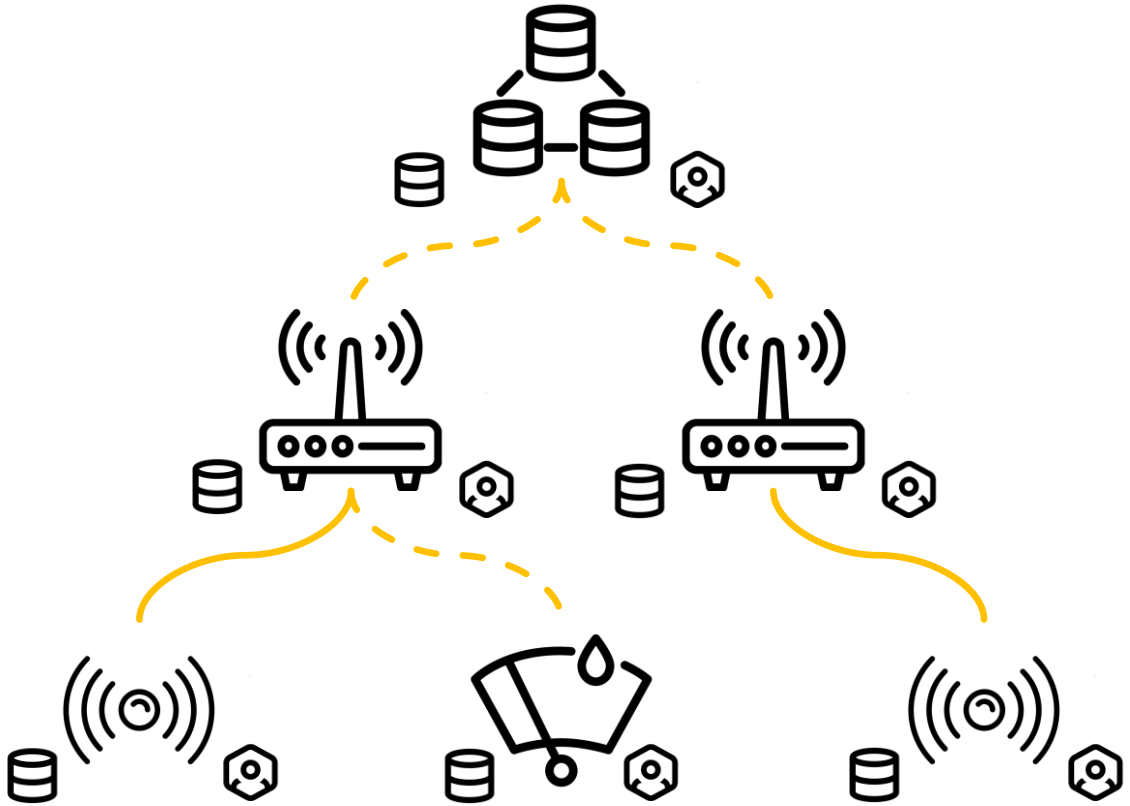
Approach to Securing IoT Network



Approach to Securing IoT Network



Approach to Securing IoT Network



Power = 3

Power = 1.5

Power = 1

Approach to Securing IoT Network

Technologies Used:

- Distributed Database
- Asymmetric Cryptography
- Symmetric Cryptography
- Public Key Identification System
- Blockchain technology encompasses all the above technologies under a single umbrella
- + an immutable record history enabling full post-incident historical analysis

Reading Material For Blockchains

How most BlockChains Work today

- You can download the software as a VM or even compile it yourself.
- You launch the code on your own servers – this makes you a “miner” as soon as the system has initialized itself.
- The system downloads the entire current BlockChain, from other machines already running the BlockChain software (there is a web site listing some you can contact for copies).

Next, you verify the BlockChain

- You'll need to recompute all the Merkle trees and the chain of hashes.
- In fact this may not take enormously long… today. Few BlockChains have huge amounts of content.
- But someday, we might have BlockChains with hundreds of billions of records and total sizes in the petabytes. Then download speed and verification time and storage will become an issue!

Meanwhile, new blocks arrive

- Each block extends some specific sequence of prior blocks.
- If you turn out to have downloaded the wrong sequence, you may have to truncate your chain and download the longer sequence.
- This is a “rollback”. During startup, substantial rollbacks can occur. Later they shouldn’t (assumes a fully connected network of mostly “correct” miners).

At this point you can create transactions

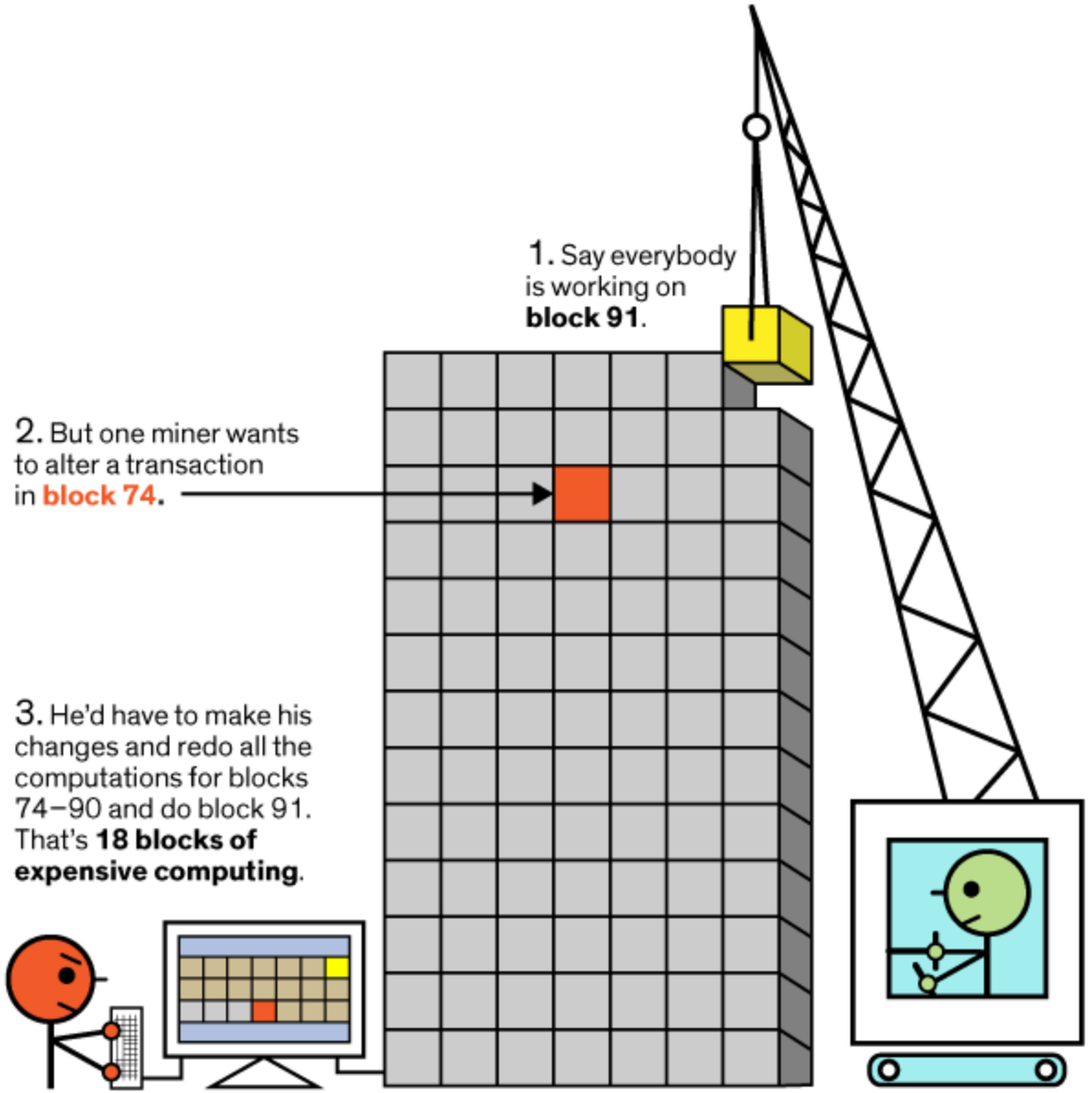
- So, you open for business.
- Someone shows up to buy a glass of your fresh lemonade.
- You'll generate the transaction (think "credit card payment slip") and submit it to the system. It enters a pool of pending transactions.

When will your transaction go through?

- Within an hour or so, you should see that your transaction got included into some block, and also that everyone seems to have adopted that block.
- The chain has moved six or more blocks into the future.
- So now you can hand that glass of frosty bliss to your happy customer!

But nobody can cheat!

- To modify a past record you need to also modify every signature subsequent to that record.
- The step where you have to find these nonce values will be very slow and you'll lose the race.



https://medium.com/loom-

http://www.cs.cornell.edu/courses/cs5412/2019sp

What if the attacker is a country?

- A country could build whole datacenters, equipped with hardware to compute SHA-256 at ultra-high speeds.
- In this case P (using the datacenter) could generate a lot of blocks quickly, for which they would be paid. Or could have an entire second BlockChain starting from months ago, and *longer than the official main one*.
- To prevent most such attacks, BlockChain solutions make the proof-of-work task harder as a function of the rate at which blocks are being found.

What if the attacker is a country?

- In effect, if P controls enough computing power, he can “gain control” of the BlockChain. The proof-of-work can become so hard that only P has the compute power to solve the puzzle!
- P could then refuse to post some transactions, or cause trouble in other ways.
- But this form of attack has not (yet) been seen.

What about races?

- Permissionless BlockChains are at risk of a “race” situation in which one group of miners is working to append record R, and some other group, record S. A tie can easily occur.
- BlockChain systems “adopt the longest chain” (may the best miners win). This can cause a rollback if a few blocks were appended by group A, but then group B suddenly publishes a longer extension.
- In practice, rollbacks longer than 6 blocks are never observed.

In contrast, Permissioned BlockChain doesn't need Proof of work

- A permissioned system is operated by known, trusted, authorized servers.
- They won't attack the chain by trying to overload it with transactions in an unfair way, and they would charge for any transactions they append on behalf of external clients.
- So we can avoid this costly step with datacenter BlockChain solutions.

What if you don't really trust the permissioned provider?

- We can mix methods: a global "proof" with a local "data store"
- Our permissioned provider can commit to some form of cryptographic root of each new version of the log (or tree), and to a proof that the new version extends the old version.
- The "commit" is broadly shared and pins the provide down. Then for an append or a query, the provider can be asked to also provide a proof that they did the append, or that the query response is correct & complete

What's In a Transaction?

- Some BlockChain systems are very rigid. For example, a BitCoin BlockChain record can only support a few operations on BitCoins.
- These represent transactions: Ken sells Ittay a packet of gum for 10
- In a permissionless scheme, Ken would probably wait for a while before handing the gum to Ittay. With permissions, rollback risk can be reduced or even completely eliminated.

Fancier Transactions

- There are several standards for encoding fancier “digital contracts” into records suitable for BlockChain.
- One, called HyperLedger, uses HTML as its underlying “language”.
- A second, Ethereum, has a sophisticated language of its own, and can even encode computational tasks into the transaction record.

Could we use BlockChain for IOT?

- This is a topic generating huge interest!
- For example, in a smart farm, a BlockChain could be used as a tamperproof audit trail, proving that animals had proper vaccinations and vet checkups, tracing food they ate and other life events, and later tracing the entire food supply chain from farm to table.
- Cornell's Vegvisir BlockChain focuses on this case. Intermittent connectivity is a strength of Vegvisir: it can handle periods of disconnection.

But there are also many issues

- From the chain, how can an auditor be sure that the transactions reflect the actual farm with its actual animals and sensors, and not a “simulation” of a farm with fake information?
- What should be the requirements for this form of monitoring and auditing, and how costly will it be to perform?
- What if we don’t trust the software? What about privacy?

More issues

- Farming is big business, and operates with loans, futures contracts, conditional agreements that can play out in many ways, etc.
- Would a single chain somehow need to encode all the farm-related digital contract events in the whole world? Even with one chain, how will its resources be managed? What would we do if a portion is irretrievably lost?
- If not, if we have multiple chains, how would they be integrated?

More Issues

- With permissionless BlockChain, is it really “safe” to trust that after six blocks have been appended, the chain won’t roll back and invalidate my transaction? (“When should Ken give the lemonade to Sally?”)

- If a smart contract references future events, what would be the “semantics” of that contract, in a PL sense? Does the meaning depend on waiting for the future to occur? Can chains of dependencies arise, or contracts that are undecidable, or infeasibly complex to “evaluate”?

Aside (Time permitting)

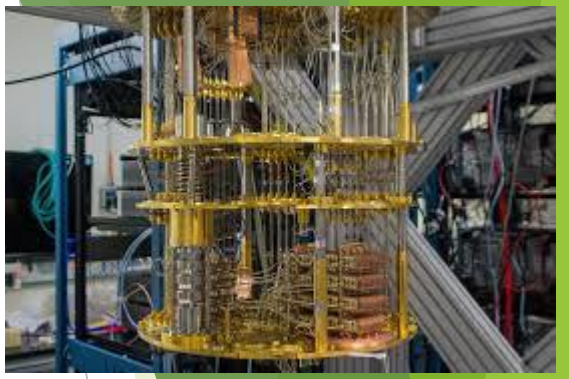
- **Will Quantum Computing really break cryptography?**

- Which is closer to the truth:
  - A quantum computer can make non-deterministic guesses, check to see if any are right (like guessing the factors of an RSA key), and then output the correct one.
  - A quantum computer can compute a near-infinite number of discrete fourier transforms “concurrently”, but you can only read out one data-point of the result at a time.

Public Misunderstanding

- Popularity of the “many worlds” interpretation of physics has clouded the public conception of what a quantum computer can do!
- In fact many worlds could be a valid model, for the most elementary level of Planck-scale physics (the layer where people talk about mbranes and string theory, and loop-quantum gravity).
- But our macroscopic (“causally emergent”) world is very remote from that most basic layer of physical reality.

Shor's Algorithm



- To factor RSA, Shor's algorithm requires a special circuit specific to the size of the keys.
- Then we input "all possible" n-bit integers, where n is the key length, like 1024. This involves a "coherent entanglement" of n qbits. But due to errors, qbits rapidly decohere. Error correction will require vastly more qbits, and nobody is sure how many. Perhaps millions or billions.]
- The entangled data is then transformed by the circuit, which computes a DFFT

Reading the output

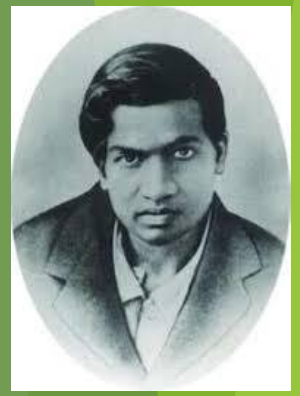
- You read the output of a quantum computer by setting up the experiment again and again and then repeatedly extracting a single sample.
- Over time, the values you read build up to a kind of probability density image, like a photo created pixel by pixel.
- In the case of Shor's algorithm this photo shows peaks that hint at the values of the factors. Now you can search for the factors close to those peaks. Quality of the search will depend on the sharpness of the peaks.

A lot of assumptions!

- Nobody knows how quantum error correction “scales”. Today it works for 3 to 5 q-bit entanglements, at best.
- Nobody knows how complex a computation we can perform without destroying coherence. In fact these quantum DFFT operations must be reversible in order to remain coherent, and hence perfectly precise.
- Nobody knows how quickly we can set up such a run and sample it.
- Nobody knows how sharp the peaks will need to be as a function of key length.

And worst of all…

Unfortunately, neither Euler nor Ramanujan really looked closely at this question!



- Nobody knows if factoring large numbers is even a “hard” problem!
- True, we lack a fast solution today. But the complexity of factoring is unknown.
- But perhaps some numerical savant will find a solution… with classical computers! The same goes for finding a nonce with the desired hashing properties to mine blocks…

The entire Edifice could collapse!

- If you bet heavily on BlockChain, you are betting that people will figure out a way to ensure that it won't yield to some kind of attack.

- But in fact this is just a bet, today.

Provably secure systems

- There is a theory of **semantic cryptography** safe against quantum attacks. It was developed by Goldwasser and Micali, who won the Turing Award for the insight.
- They proved that secure encryption schemes must be probabilistic, rather than deterministic, with many possible encrypted texts corresponding to each message.
- The **Goldwasser–Micali** (GM) cryptosystem demonstrates the idea.

Could a BlockChain Use GM Cryptographic techniques?

- At present, the GM system is too computationally slow for practical use, and also causes too much "inflation" in the size of data.
- Each bit in the data becomes a point in a very high dimensional space, leading to a billions-to-one increase in message sizes.
- But continued research may yield much more compact solutions with the same properties. A new research initiative just started on this topic.