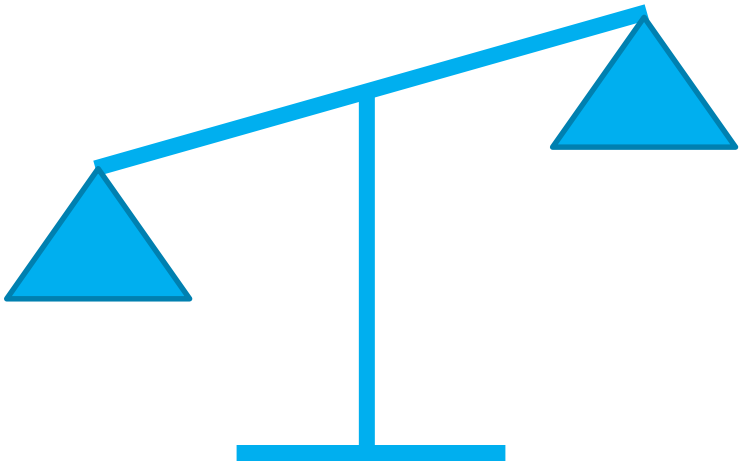
Presentation written by Marta Piekarska, Director of Ecosystem, Hyperledger and Yoram Golandsky

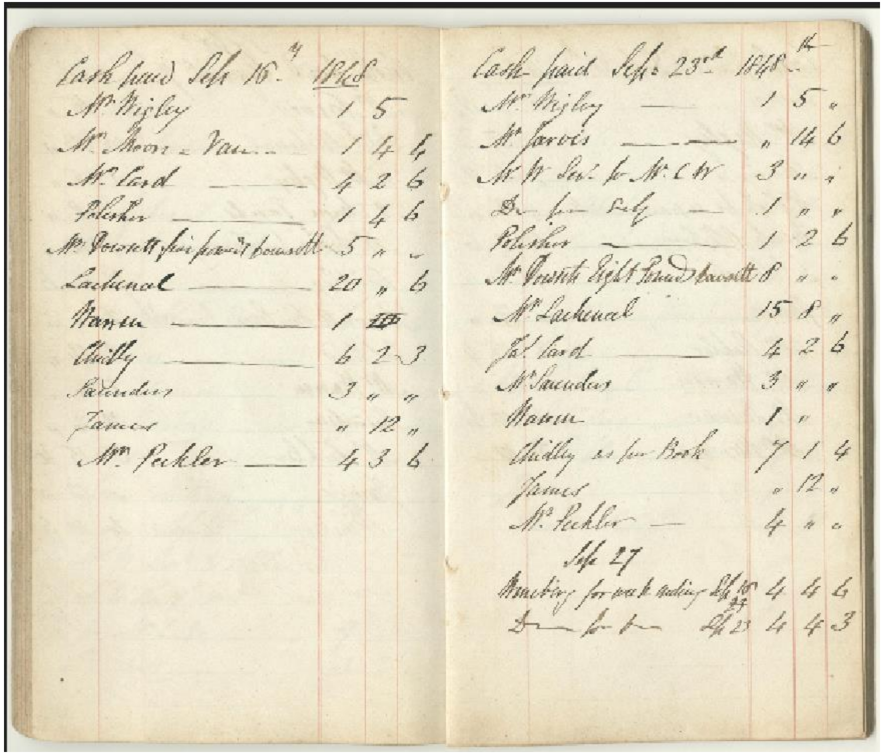
Considerations for Enterprise DLT

Speaker: Yoram Golandsky,
Board Advisor @ Wawllet (ICO)
Managing Director @ Smart Ventures

How do we agree on an assets balance?

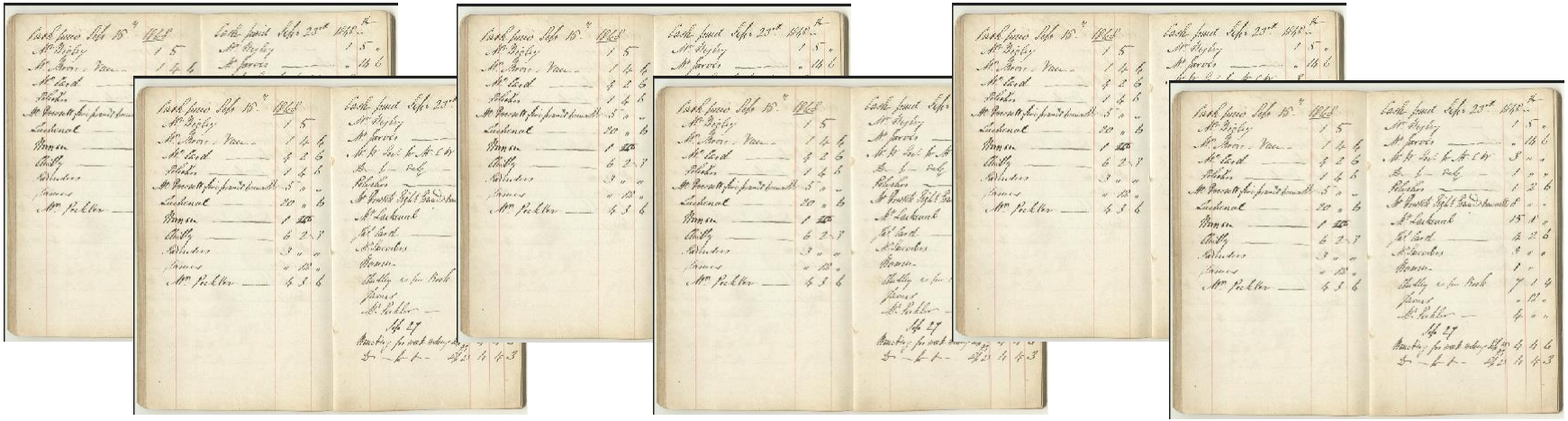


Traditional Ledgers



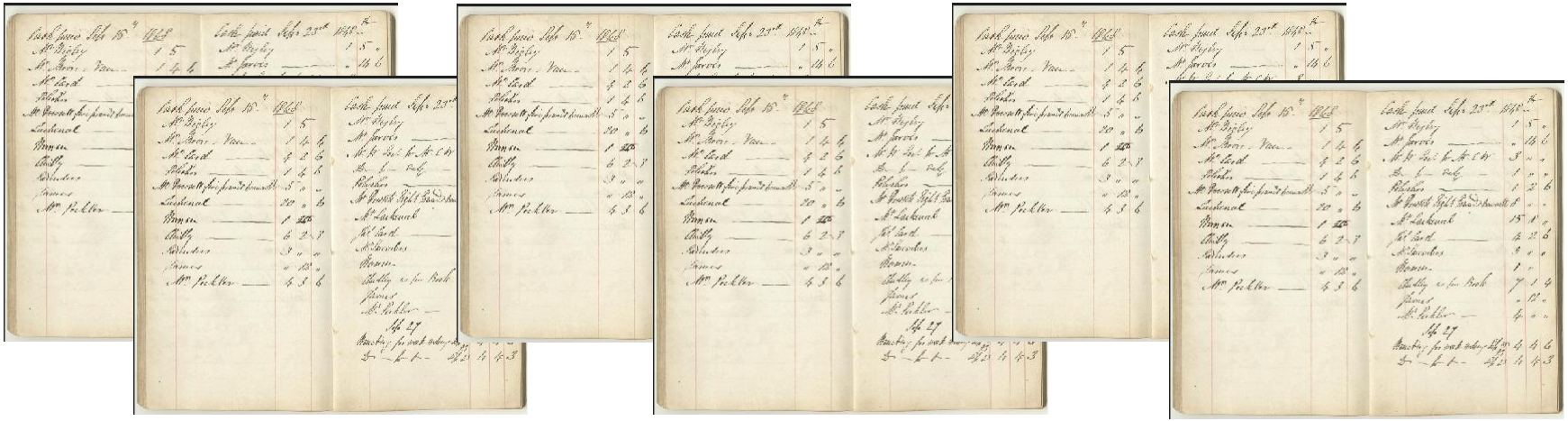
HYPERLEDGER
BLOCKCHAIN TECHNOLOGIES FOR BUSINESS

In the digital world



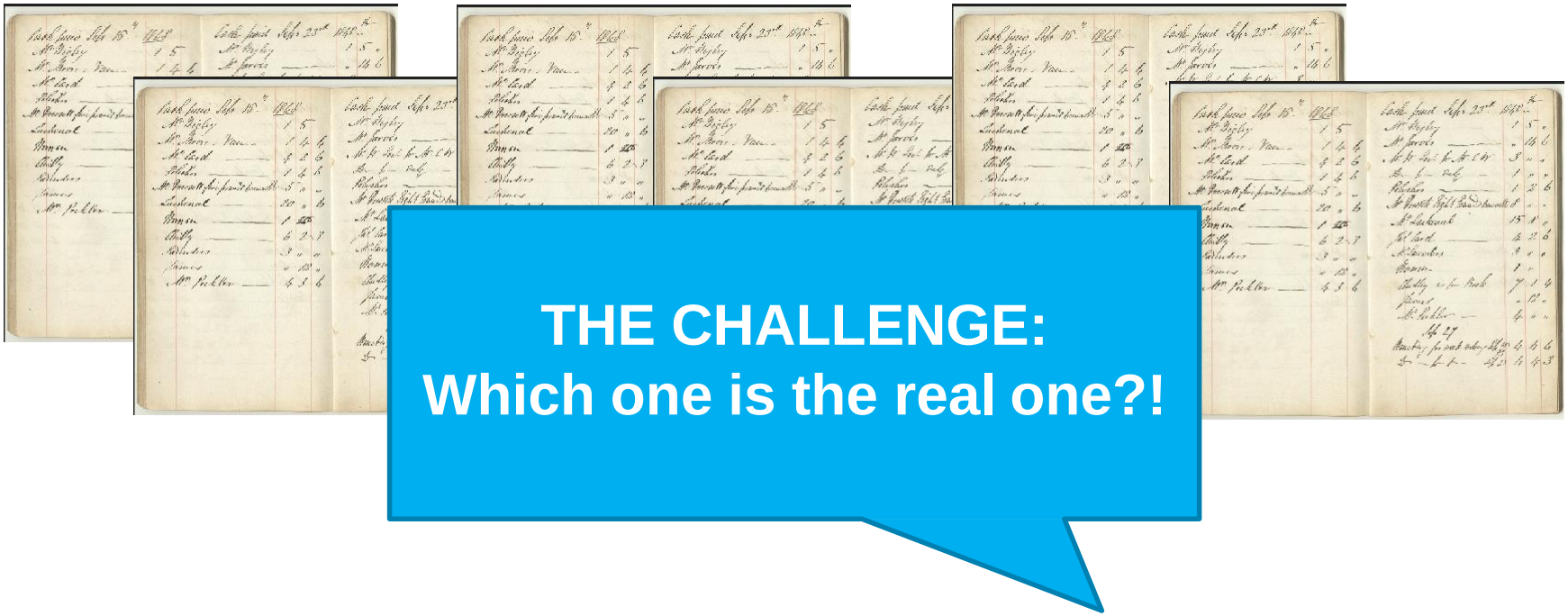
There are many copies, that may contain different versions of truth

In the digital world



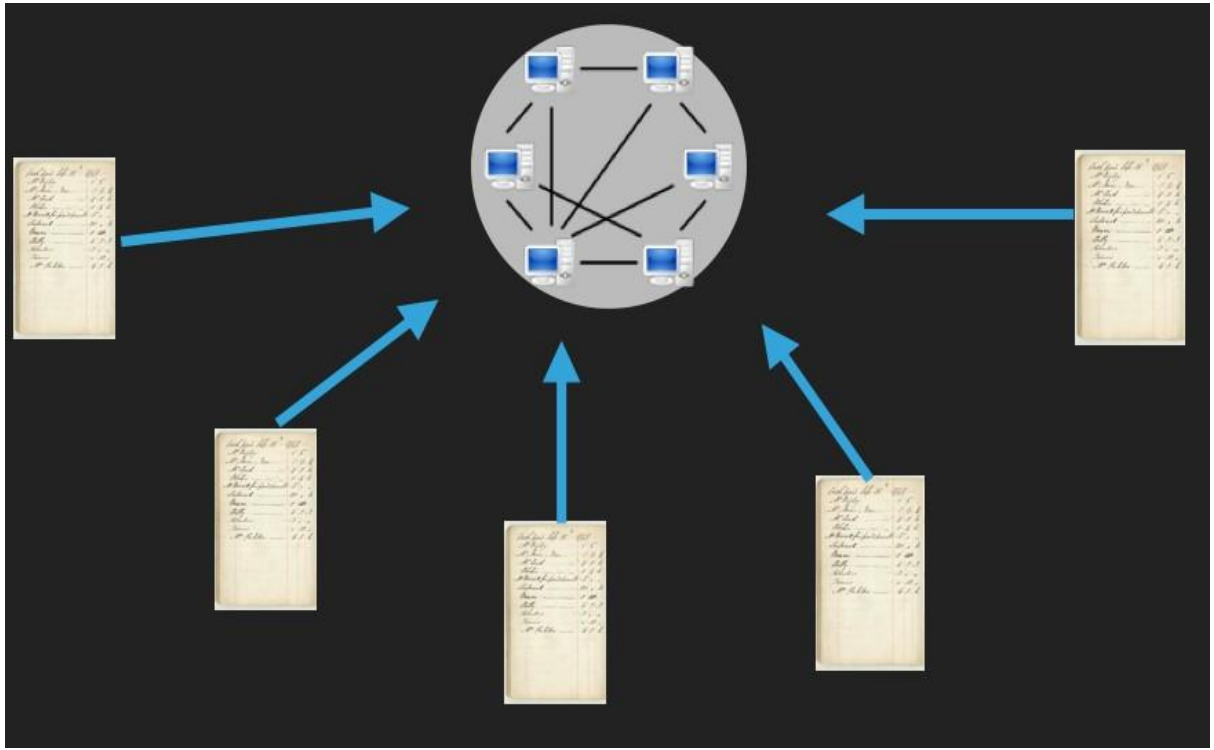
There are many copies, that may have different versions of truth

In the digital world




LET ME INTRODUCE YOU
TO THE INTERNET
WeKnowMemes

Now everyone can publish their version



Distributed Ledgers in Action

All business participating in a commercial ecosystem need a ledger to contain a record of transactions. It is vitally important to know that your copy of the ledger is identical to your business partners' and free from discrepancies.

Example scenario:

1. Everyone in a room has a book with the instructions to write down entries as they get called out.
2. Someone calls out item number one and everyone writes it down.
3. Then two people call out item number two at the same time, but the item number differs.
4. There needs to be a process for who wins, and the loser gets to try to call out item number three.
5. When all agree on the outcome of an entry, the next link in that ledger can be written.
6. Whether this happens in a small scale or the size of the internet, that is the spectrum for how a distributed ledger can work.

When Is A Blockchain Relevant?

As with any technology, it’s critical to determine the conditions under which blockchain should be applied. Blockchain can solve many, but not all, problems.

1. What are the required trust, integrity, and security levels?
2. How much data would be stored in the blockchain? Does all of the data need to be stored in the ledger or will some of it be stored outside? If off-chain storage will be used, how will this data be referenced by the transactions in the ledger?
3. How large is the business network? Will each network participant be able to access or store all information? Do the transactions need to be partitioned across the network?
4. What are the performance and transaction processing requirements of the network?
5. Is there a need to automate business processes across the network? How complex is the automation logic?

Key Considerations

The Right Business Use Case

The use cases identified for DLT need to reflect a few foundational themes

1. Non-reliance on a middleman, a business process supporting a truly distributed deployment
2. Building trust among a large number of actors/counterparties
3. Support distributed consensus, and transparency.

It’s important that a target use case be realistic from both a functional requirement standpoint as well as from a business process understanding.

Non Functional Requirements

The current state of DLT platforms is that they fall short in a few key areas that enterprises usually take for granted in other platforms.

These include key areas such as:

- data privacy,
- transaction throughput,
- high speed of
- performance etc.

The good news is that the DLT community are acutely aware of enhancements that need to be done to the underlying platforms (e.g reduced block size etc) to increase throughput

Enterprise Integration Requirements

There are very few standards and guidance on integrating distributed applications (Dapps) custom built for DLTs with underlying enterprise assets.

- These assets include:
- enterprise middleware stacks,
- identity management platforms,
- corporate security systems,
- application data silos,
- BPM (Business Process Management).

Interoperability is still in its infancy despite vendor claims

Smart Contracts are Still In Their Infancy…

The blockchain introduces the important notion of programmable digital instruments or contracts.

Smart Contract is a program that can perform the generation of downstream actions when appropriate conditions are met.

Smart Contracts are being spoken about as the key functionality for any DLT platform based on Blockchain.

Smart contracts are again not standards based across major DLT platforms. Which means that they're not auditable & verifiable across both local and global jurisdictions or when companies use different underlying commercial DLTs.

Security and Data Privacy

The promise of the original blockchain platform which ran Bitcoin was very simple.
It provided a truly secure, trustable and immutable record on which any digital asset could be run.

Parties using the system were all in a permissionless mode which meant that their identities were hidden from one another and from any central authority.

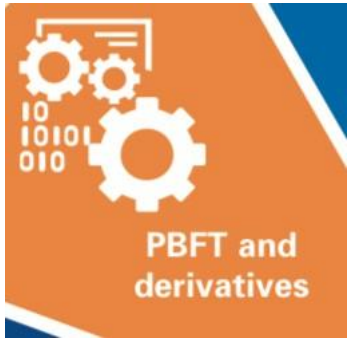
While this may work for Bitcoin like projects, the vast majority of industry verticals will need strong legal agreements and membership management capabilities which follow them.

Accordingly, these platforms will need to be permission-ed.

What makes a good consensus?

- **Decentralized governance:** A single central authority cannot provide transaction finality.
- **Integrity:** It enforces the validation of the transaction integrity (e.g., mathematically through cryptography).
- **Nonrepudiation:** There are ways to verify the supposed sender really sent the message.
- **Privacy:** There should be ways to ensure that only the intended recipient can read the message.
- **Fault tolerance:** The network operates efficiently and quickly, even if some nodes or servers fail or are slow.
- **Performance:** It considers throughput, liveness, scalability, and latency

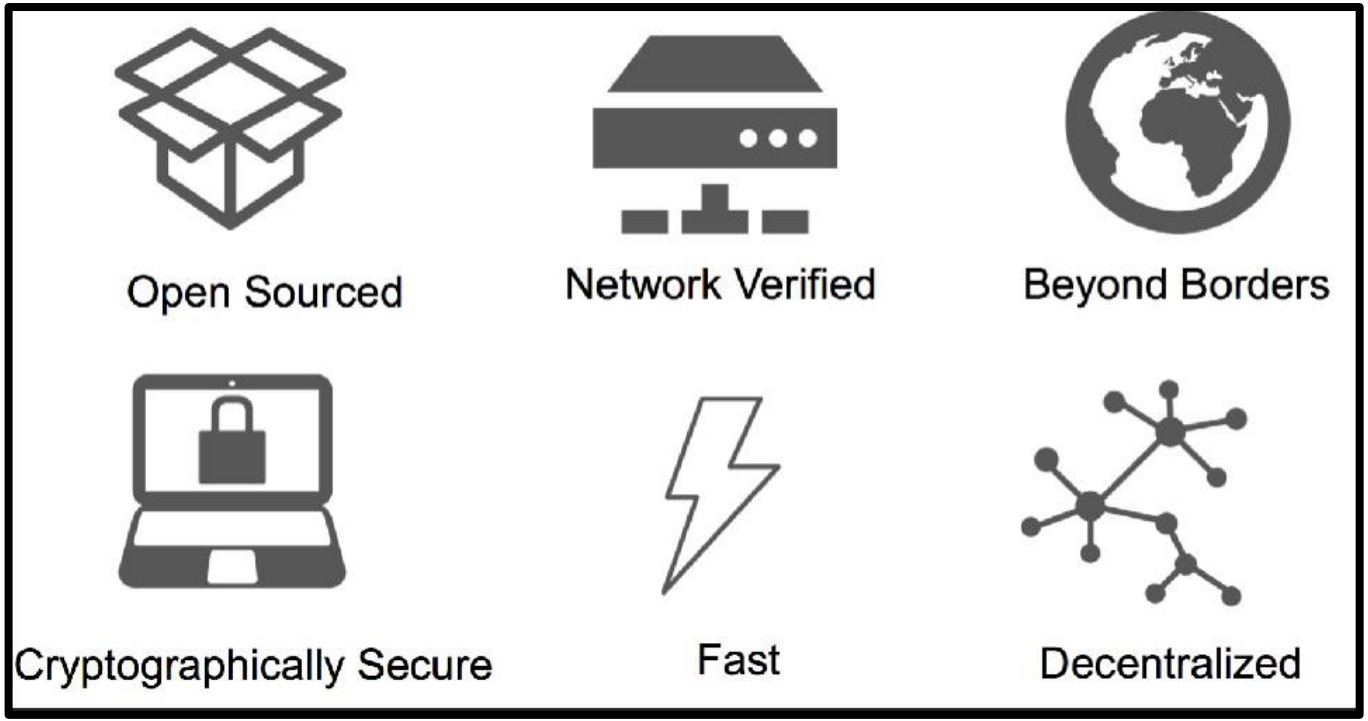
Still you need to agree on a common version



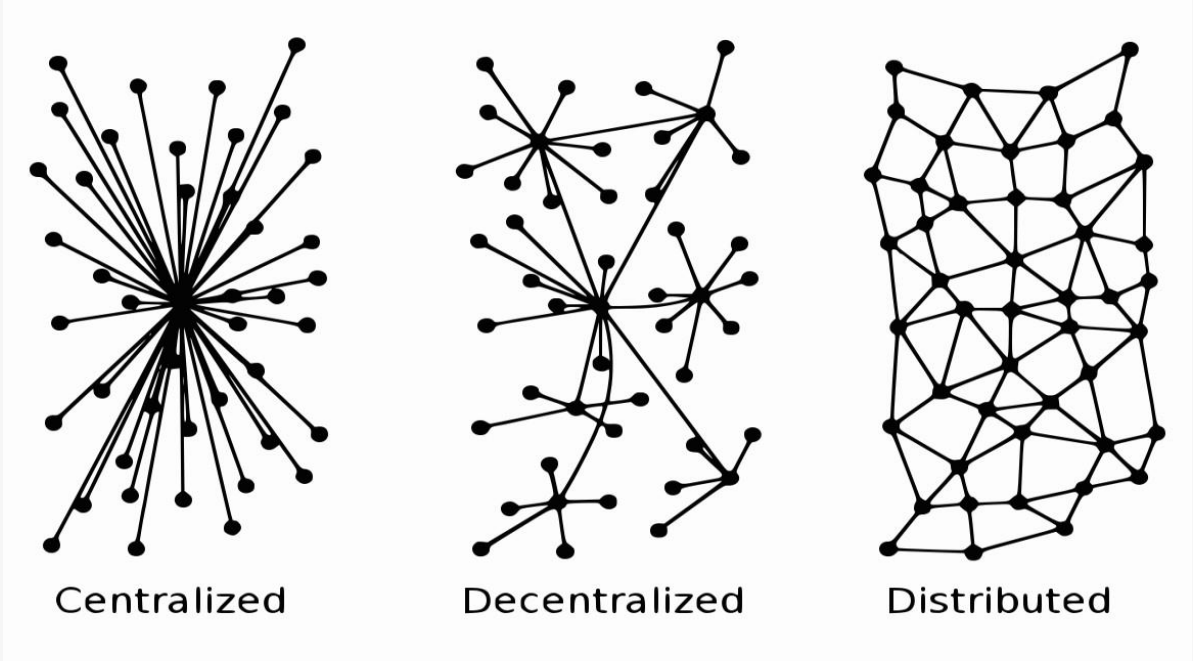
Consensus Algorithm

Characteristic	Ethereum	Hyperledger Fabric	R3 Corda
Description of platform	– Generic blockchain platform	– Modular blockchain platform	– Specialized distrib-uted ledger platform for financial industry
Governance	– Ethereum developers	– Linux Foundation	– R3
Mode of operation	– Permissionless, public or private[4]	– Permissioned, private	– Permissioned, private
Consensus	– Mining based on proof-of-work (PoW) – Ledger level	– Broad understand-ing of consensus that allows multiple approaches – Transaction level	– Specific understand-ing of consensus (i.e., notary nodes) – Transaction level
Smart contracts	– Smart contract code (e.g., Solidity)	– Smart contract code (e.g., Go, Java)	– Smart contract code (e.g., Kotlin, Java) – Smart legal contract (legal prose)
Currency	– Ether – Tokens via smart contract	– None – Currency and tokens via chaincode	– None

Finally, the agreed block is added to the ledger and distributed



The less trust required the lower the maintenance cost



Myth Debunked: Blockchain ≠ Cryptocurrency

Cryptocurrency is an application that sits on top of blockchain.

Not the other way around.

Blockchain and cryptocurrencies are often discussed in similar contexts, but they are <u>not</u> one in the same.

What is a Smart Contract?

The code or any complex program stored and executed on a blockchain.

Building one layer above distributed ledger technologies, you have smart contracts. Since distributed databases allow for multiparty, shared database use, distributed ledgers can be equipped with multi-party business logic, which is more commonly referred to as ‘smart contracts. These are a way to publish scripts and write programs in a language particular to a given chain, have them live on the shared network, and be able to execute on all those different nodes simultaneously.

Smart Contracts in Action

Imagine a farmer based in Sacramento, California buys an insurance agreement that protects them from extreme weather condition. If temperatures reach more than 100 degrees for 100 days, they get reimbursed 10,000 USD.

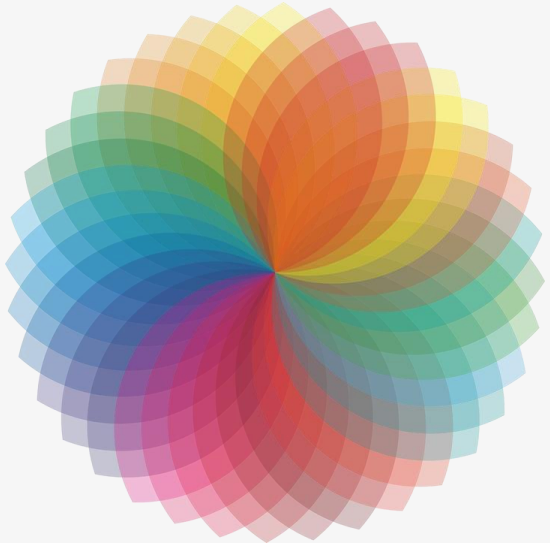
With human ledgers widely operated today, the insurer might find a way to back out of, procrastinate or dispute this agreement.

If a Smart Contract is in place, the script in the ledger would rule that on that 100th day of 100+ degrees, the 10,000 USD would be automatically withdrawn. With an automated process, like it or not, the insurer cannot back out.

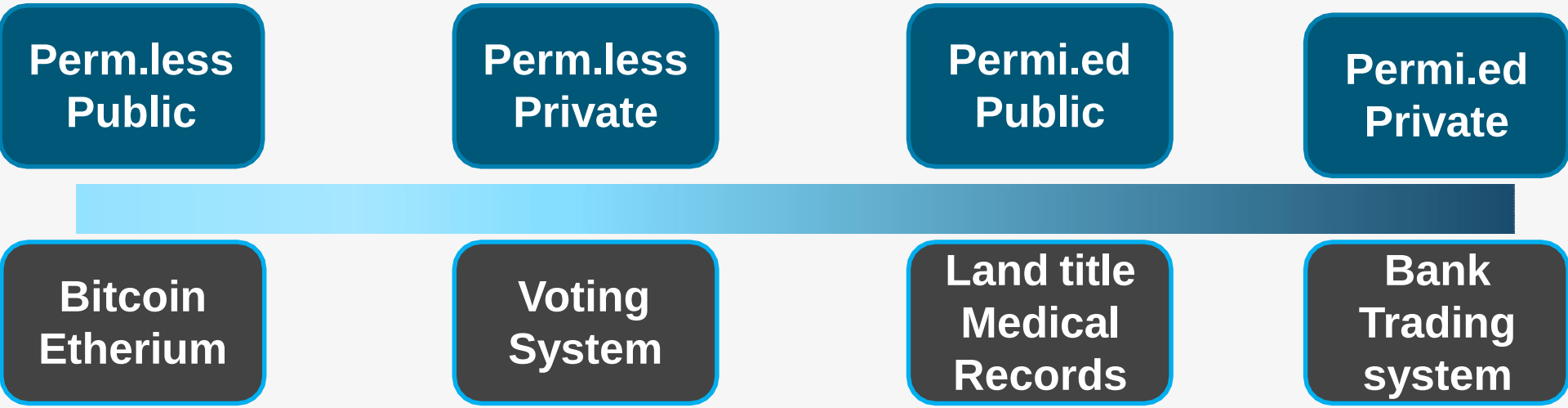
Why Business Blockchain Technologies

All over the global market there are ledgers that organizations and individuals alike must trust.

Permissioned blockchains differ from what you'll find in Bitcoin or Ethereum.



Permissioned? Private?



- **Permisioned vs Permissionless: Who can Write to a blockchain (ACCESIBILITY)**
- **Public vs Private: Who can read from a blockchain (VISIBILITY)**

Permissionless Blockchains

Characteristics:

- Participation open to the **public**
- **Peer-to-peer** transactions
- Typically tied to **cryptocurrency**
- Fully **decentralized**

Challenges:

- **Privacy** and **scaling**

Permissionless blockchains are a disruptive technology that can dramatically change how we conduct business activities.

Permissioned Blockchains

Characteristics:

- **Private**
- **Controlled**
- **Trusted**
- **Efficient**
- **privacy**
- **confidentiality**

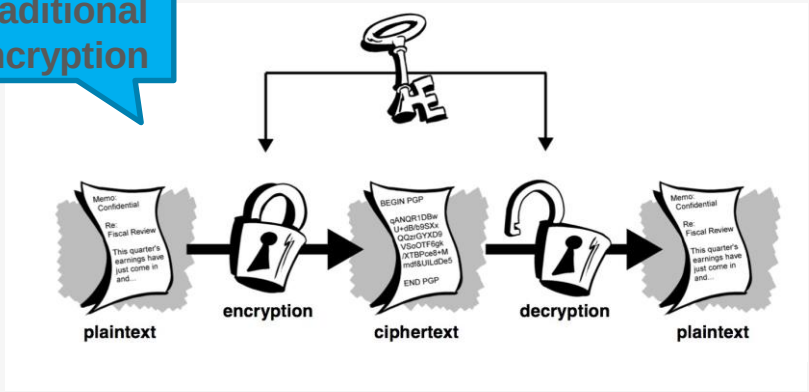
Challenges:

- Some level of **centralized trust** through governing authority

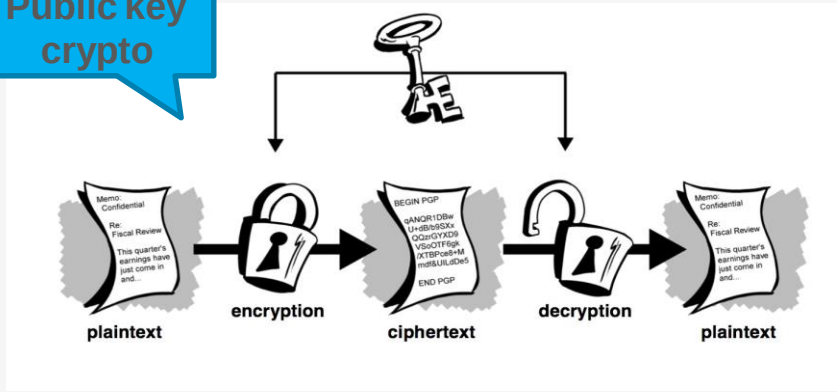
Permissioned blockchains may lead to cost-savings, workflow improvements, automation and improved auditing with current business processes.

A bit of background in Crypto

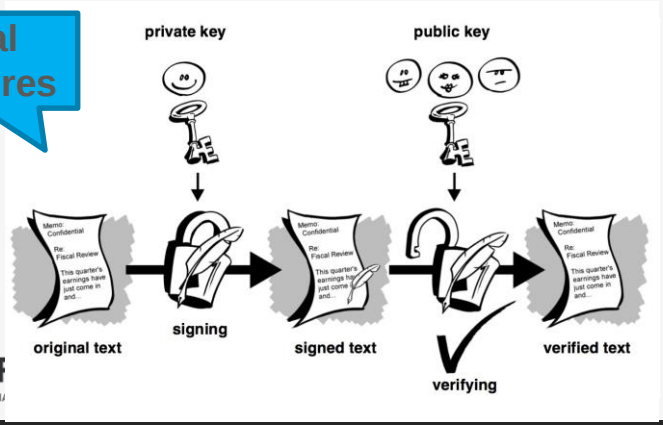
Traditional encryption



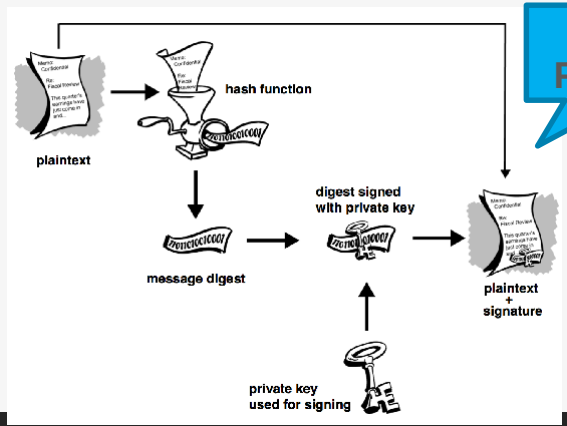
Public key crypto



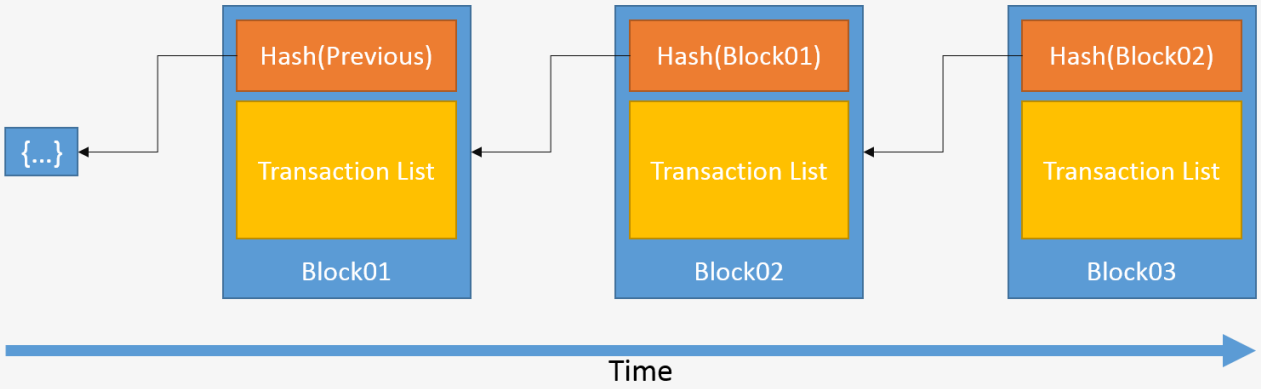
Digital Signatures



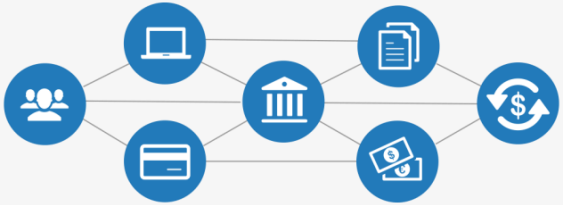
Hash Functions



Blockchain, trust and transparency



Early Adopter Industries

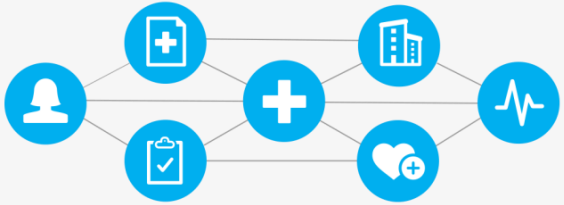


Financial Services

The most immediate business opportunities are to use business blockchains to track and trade stocks and bonds with reduced risk and time, and increased transparency.

Supply Chain

Blockchain technologies will lead to greater efficiency, as well as safer and more ethical standards through the ability to prove the veracity of transactions to all stakeholders.



Healthcare

Business blockchains are giving the healthcare industry a chance to reinvent what historically has been a thorny problem: sharing patient data records between organizations.

Hyperledger is a collaborative and global open source software community, hosted by The Linux Foundation, advancing blockchain technologies for business.

Hyperledger Momentum

<2
years since launch

25k+
Commits

4
Tools

5
Frameworks

1
Production Release
Hyperledger Fabric v1.0

180+
Members
(30+ in China)

12
Active Community Working Groups

80
Meetups Worldwide

16k
Meetup Participants

600+
Media Clips Per Month in 2016

HYPERLEDGER
BLOCKCHAIN TECHNOLOGIES FOR BUSINESS

The Hyperledger Umbrella

Infrastructure

Technical, Legal, Marketing, Organizational

Ecosystems that accelerate open development and commercial adoption

THE LINUX FOUNDATION

Cloud Foundry | Node.js | **Hyperledger** | Open Container Initiative

Frameworks

Meaningfully differentiated approaches to business blockchain frameworks developed by a growing community of communities

Hyperledger **Indy** | Hyperledger **Fabric** | Hyperledger **Iroha** | Hyperledger **Sawtooth** | Hyperledger **Burrow**

Tools

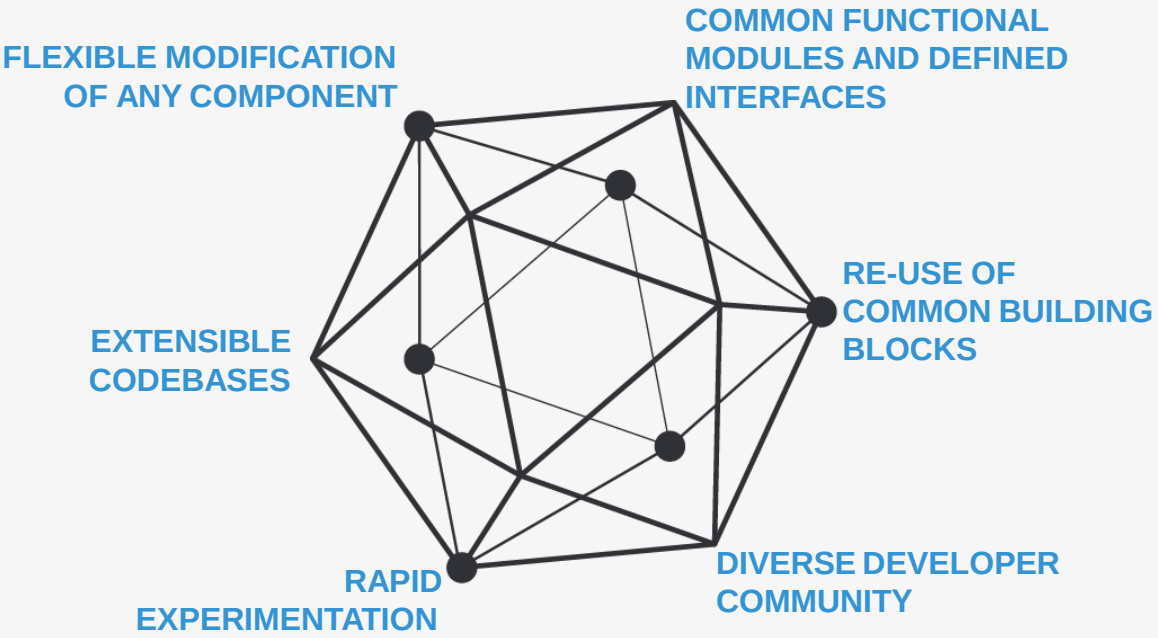
Typically built for one framework, and through common license and community of communities approach, ported to other frameworks

Hyperledger **Composer** | Hyperledger **Explorer** | Hyperledger **Cello** | Hyperledger **Quilt**

Hyperledger Blockchain Frameworks

Hyperledger Fabric	Intended as a foundation for developing applications or solutions with a modular architecture, Hyperledger Fabric allows components, such as consensus and membership services, to be plug-and-play.
Hyperledger Iroha	A business blockchain framework designed to be simple and easy to incorporate into infrastructural projects requiring distributed ledger technology.
Hyperledger Sawtooth	A modular platform for building, deploying, and running distributed ledgers. Hyperledger Sawtooth includes a novel consensus algorithm, Proof of Elapsed Time (PoET), which targets large distributed validator populations with minimal resource consumption.
Hyperledger Burrow	A permissionable smart contract machine. The first of its kind when released in December, 2014, Burrow provides a modular blockchain client with a permissioned smart contract interpreter built in part to the specification of the Ethereum Virtual Machine (EVM).
Hyperledger Indy	Tools, libraries, and reusable components for providing digital identities rooted on blockchains or other distributed ledgers so that they are interoperable across administrative domains, applications, and any other silo.

Hyperledger Modular Approach Benefits



Hyperledger Goals

Where open source teams build diverse approaches for business blockchain technology systems

Create enterprise grade, open source, distributed ledger frameworks & code bases

to support business transactions

Provide neutral, open, & community-driven infrastructures

supported by technical and business governance

Build technical communities

to develop blockchain and shared ledger POCs, use cases, field trials and deployments

Educate the public

about the market opportunity for blockchain technology

Promote our community of communities

taking a toolkit approach with many platforms and frameworks