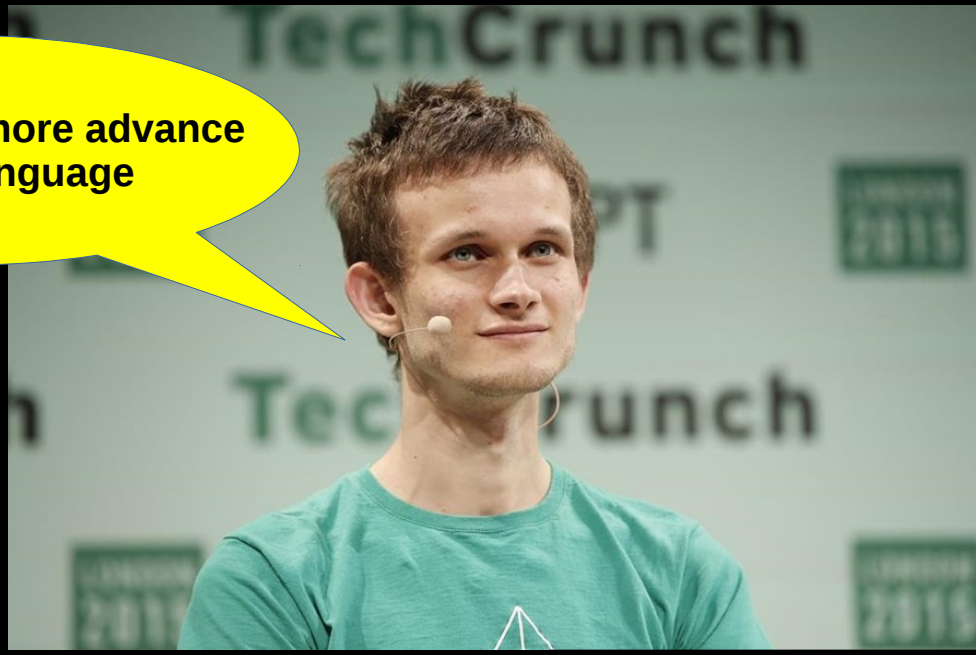
Lecture 3

Ethereum
&
Hyperledger

Ethereum



Vitalik Buterin

1. Almost like Bitcoin
2. Have Smart Contracts

Smart Contracts

- Written in Solidity.
- scripting language specially created for Ethereum

Smart Contracts

Live example…

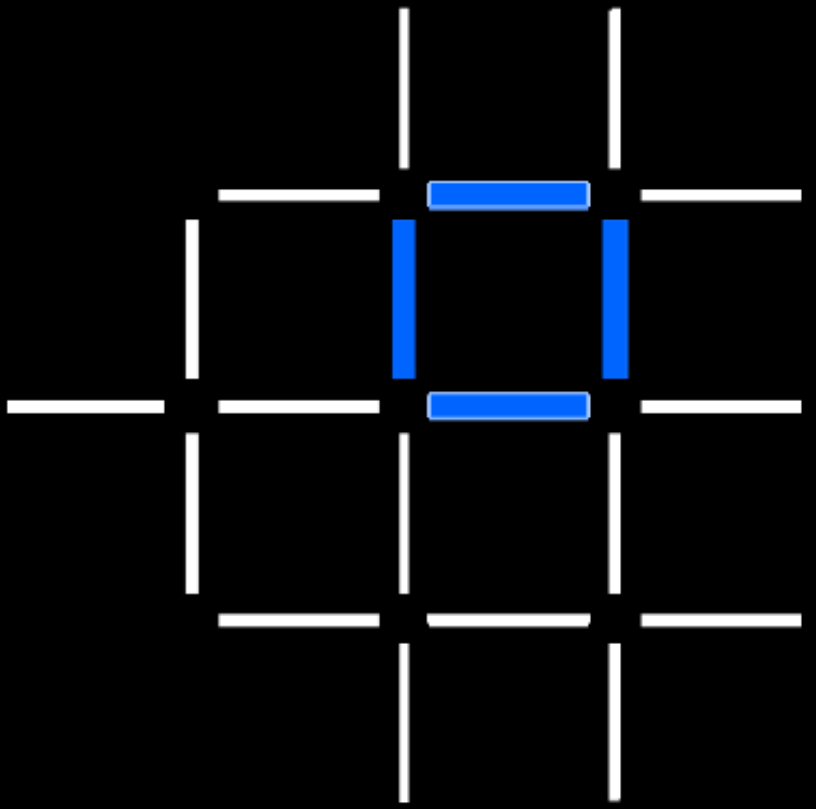
https://remix.ethereum.org

Hyperledger

Blockchain Overview

Unit 01

DRAFT – May 15, 2018

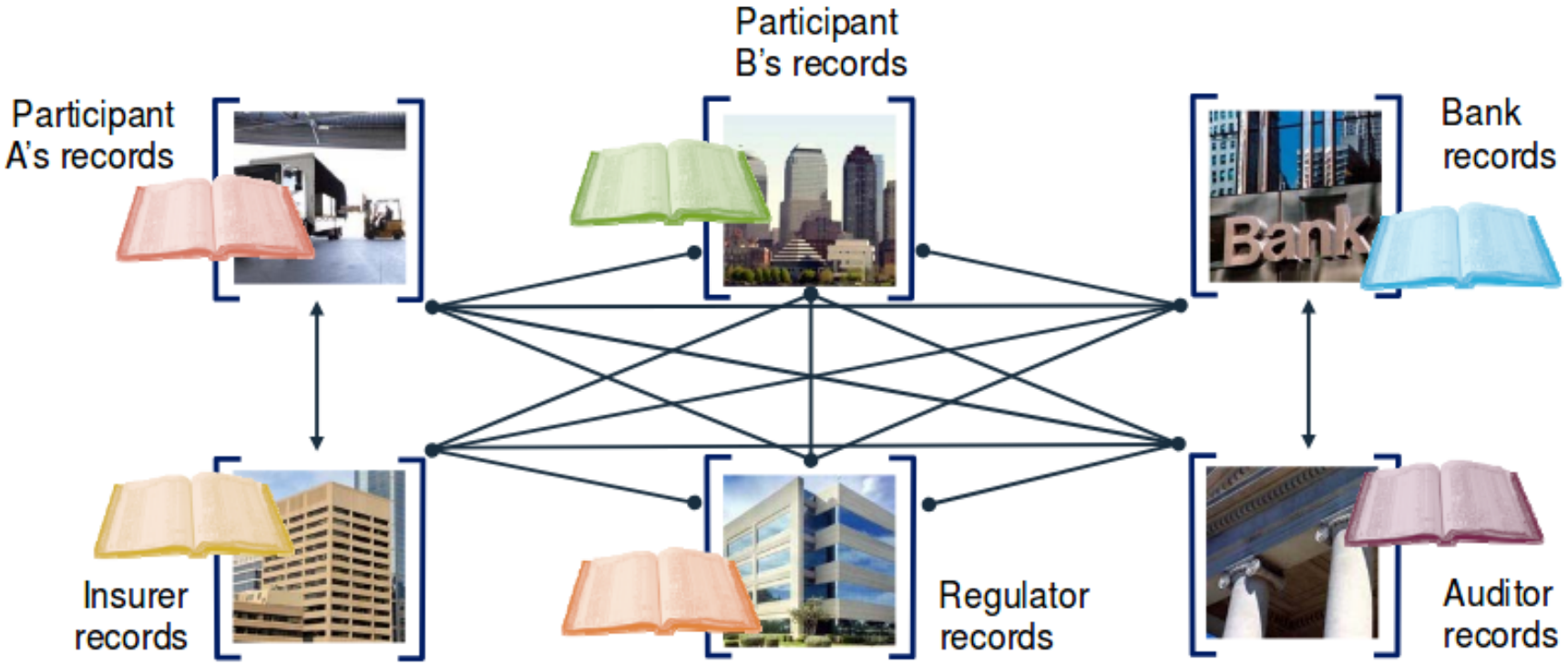


DRAFT v1.0 May 2018

IBM **Blockchain**

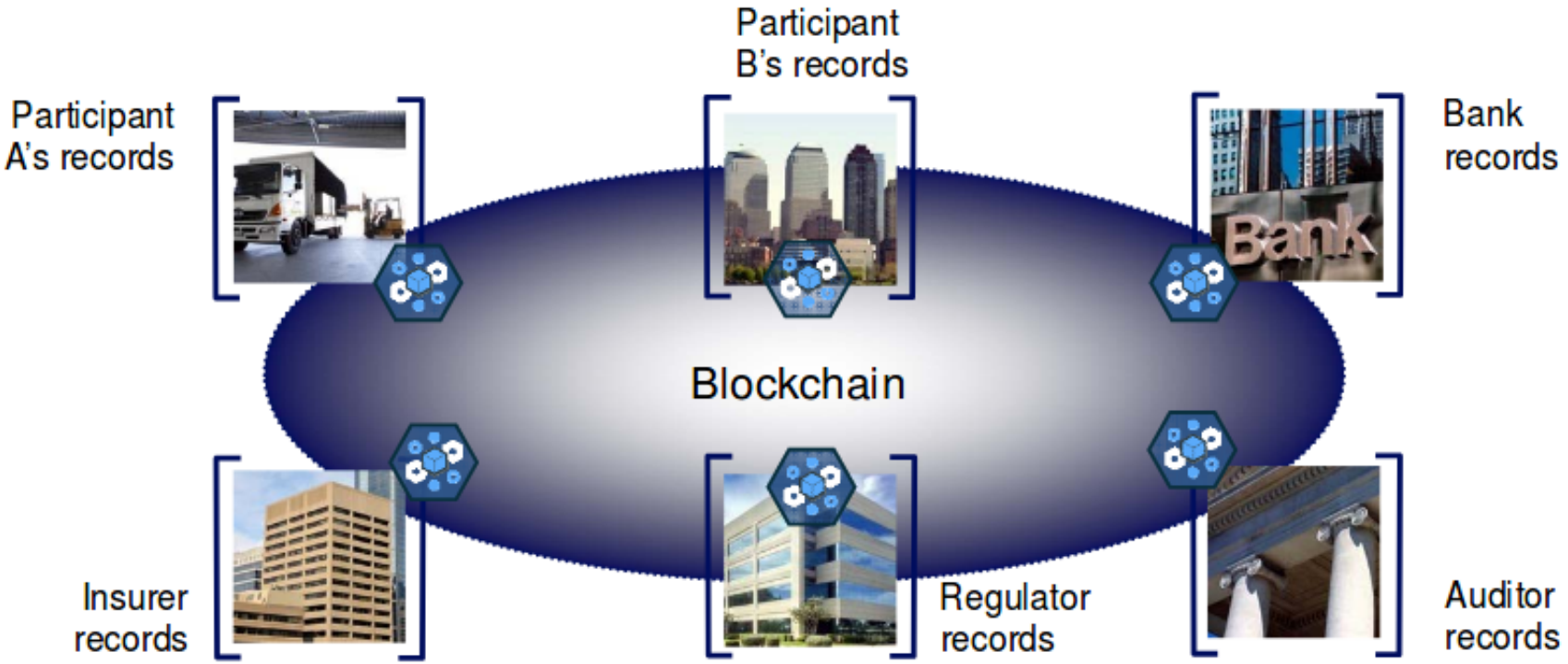
IBM

Problem…



… inefficient, expensive, vulnerable

A shared, replicated, permissioned ledger …

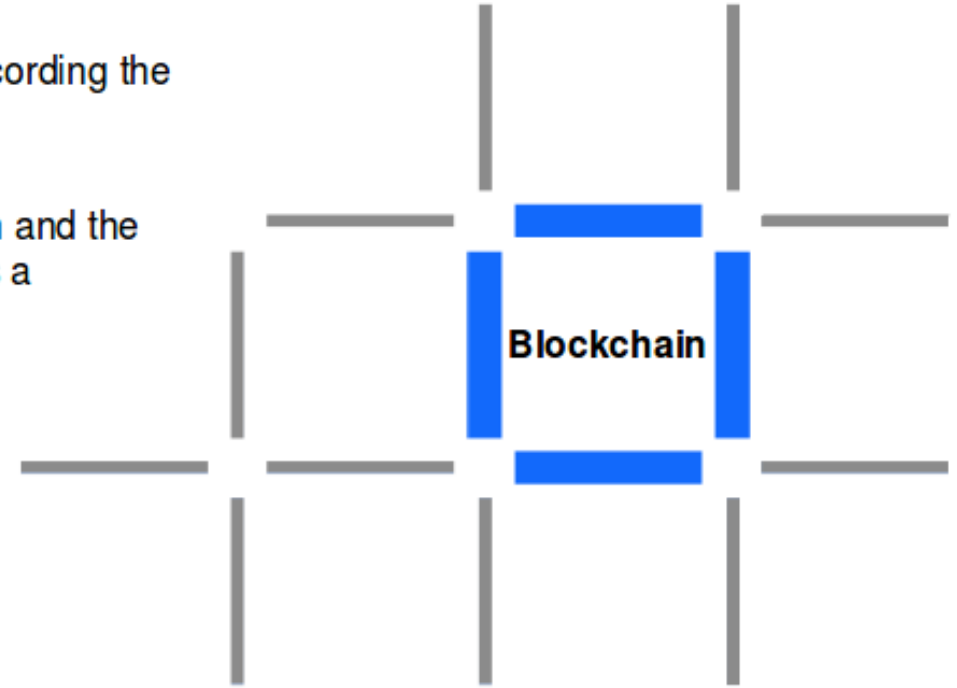


… with consensus, provenance, immutability, and finality

What is blockchain?

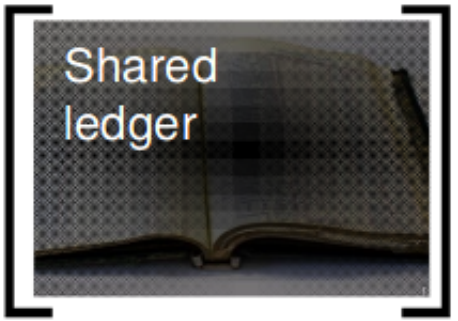
Blockchain is a **shared immutable ledger** for recording the history of transactions.

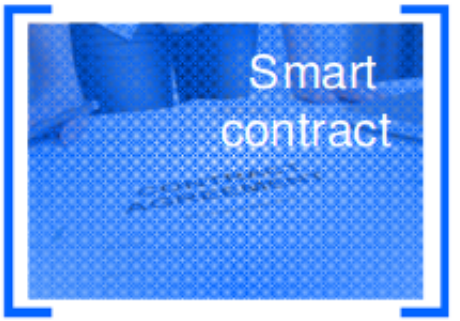
A **business blockchain**, such as IBM Blockchain and the Linux Foundation's Hyperledger Project, provides a **permissioned network** with known identities.



Blockchain for business requirements

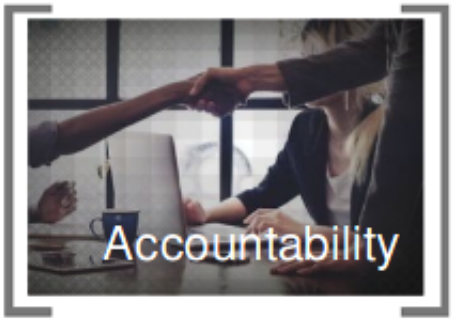
Append-only distributed system of records shared across business networks





Business terms executed with transactions

Transactions are secure with appropriate visibility

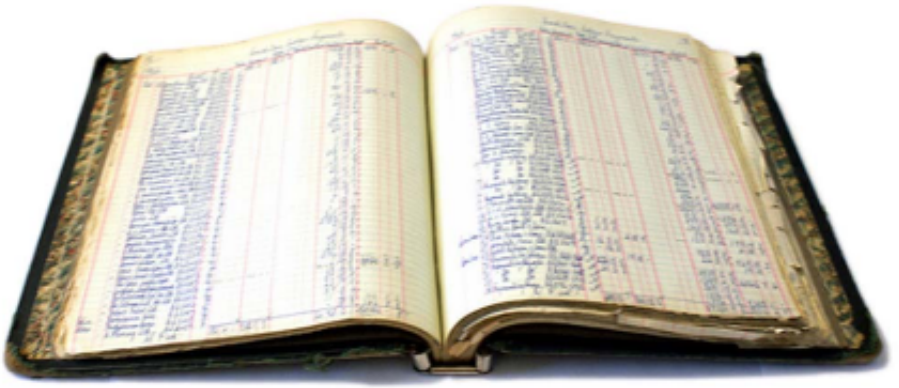


Transactions are provably endorsed by relevant participants

Shared ledger

- Shared between participants
- Participants have own copy through replication
- Permissioned, so participants see only appropriate transactions
- THE shared system of record
- Immutable due to an append-only data structure

Records all transactions across business networks



Smart contract

- Verifiable, signed
- Business rules, written in programming languages, supported by the blockchain technology
- Examples:
  - Defines contractual conditions under which a bond transfer occurs
  - Defines rules on which a vehicle can be transferred to a new owner

Business rules associated with the transaction

Privacy

- Participants require:
  - Appropriate **privacy** and **confidentiality** between subsets of participants
  - Identity not linked to a transaction
- Transactions need to be authenticated
- Cryptography is central to these processes

The ledger is shared, but participants require privacy and confidentiality

Accountability

- Participants endorse transactions
  - **Consensus**: Participants agree that a transaction is valid
    - Business network decides who will endorse transactions
    - Endorsed transactions are added to the ledger with appropriate confidentiality

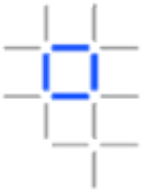
The ledger is a provable source of information

Accountability (continued)

- Assets have a verifiable audit trail
  - **Provenance**: Participants know where the asset came from and how it's ownership has changed over time
  - **Immutability**: No participant can tamper with a transaction once it is agreed upon
    - Transactions can not be modified, inserted or deleted
  - **Finality**: Only one place to determine the ownership of an asset or completion of a transaction (the shared ledger).

The ledger is a provable source of information

Bitcoin versus blockchain for business



- Bitcoin utilizes an un-permissioned public ledger:
  - Defines an unregulated shadow-currency
  - The first blockchain application
  - Resource-intensive

- Blockchains for business are generally permissioned and private, and prioritize:
  - Identity over anonymity
  - Selective endorsement over proof of work
  - Assets over cryptocurrency

Blockchain technology is the infrastructure, upon which blockchain applications are built

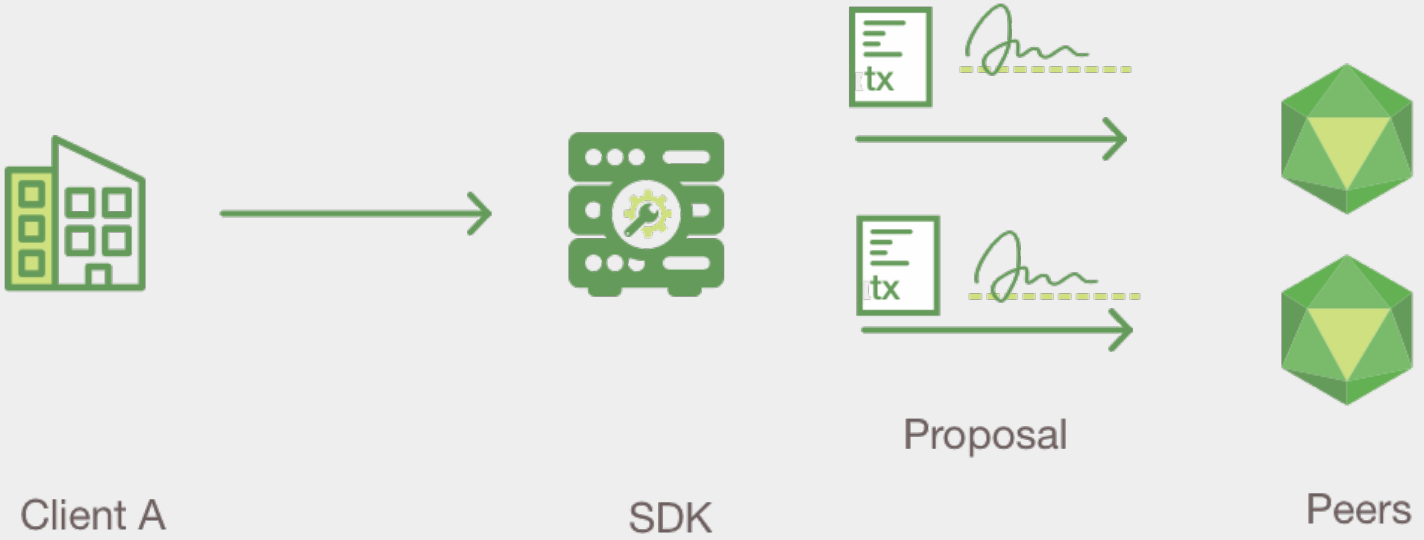
Hyperledger

Transaction Flow

Assumptions

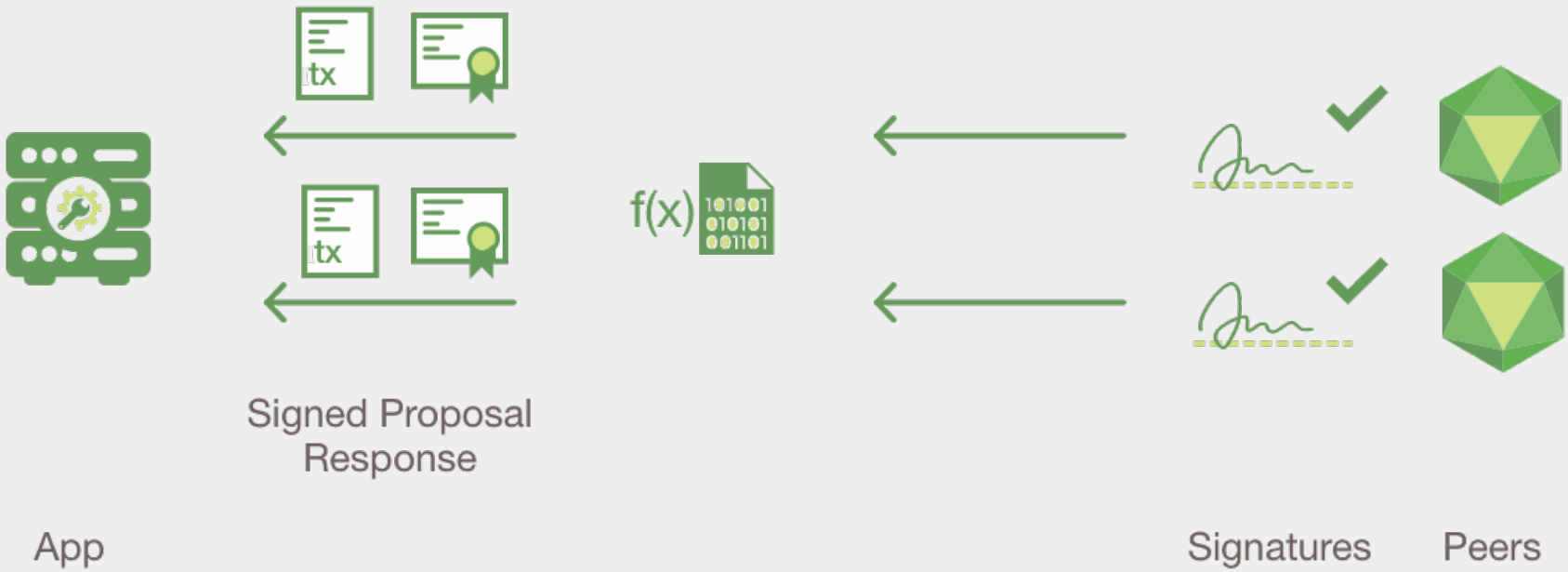
https://hyperledger-fabric.readthedocs.io/en/release-1.2/txflow.html

Client A initiates a transaction



https://hyperledger-fabric.readthedocs.io/en/release-1.2/txflow.html

Endorsing peers verify signature execute the transaction



https://hyperledger-fabric.readthedocs.io/en/release-1.2/txflow.html

How we set Endorsing peers

For example:

- `AND('Org1.member', 'Org2.member', 'Org3.member')` requests 1 signature from each of the three principals
- `OR('Org1.member', 'Org2.member')` requests 1 signature from either one of the two principals
- `OR('Org1.member', AND('Org2.member', 'Org3.member'))` requests either one signature from a member of the `Org1` MSP or 1 signature from a member of the `Org2` MSP and 1 signature from a member of the `Org3` MSP.

https://hyperledger-fabric.readthedocs.io/en/release-1.2/txflow.html

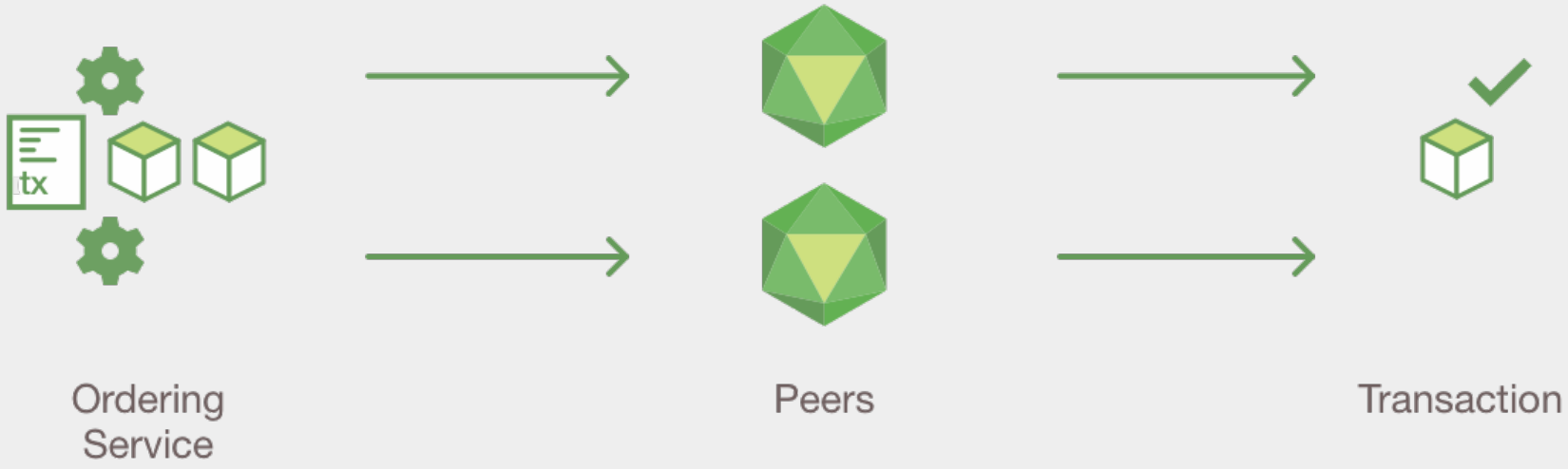
Proposal responses are inspected

https://hyperledger-fabric.readthedocs.io/en/release-1.2/txflow.html

Client assembles endorsements into a transaction

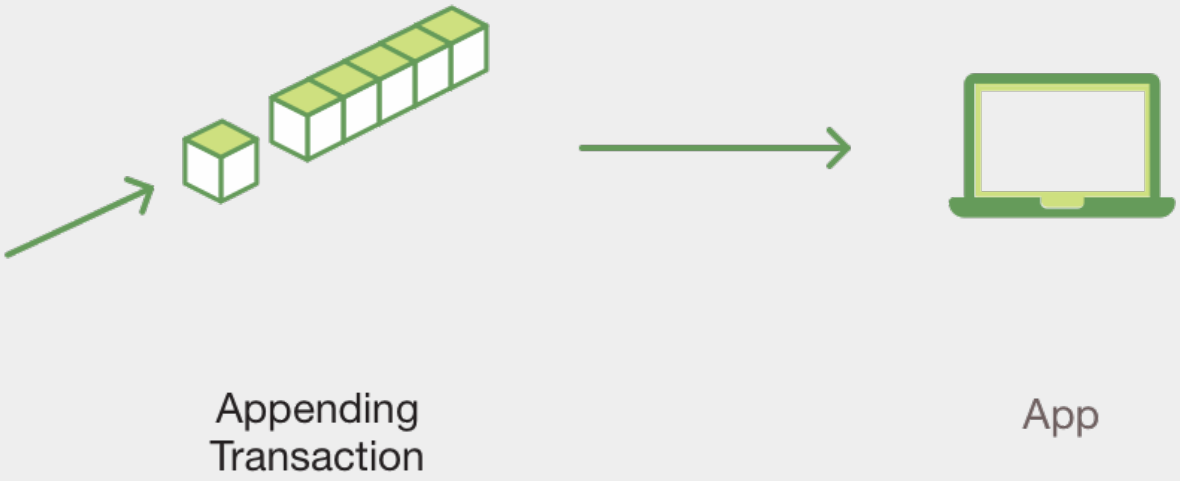
https://hyperledger-fabric.readthedocs.io/en/release-1.2/txflow.html

Transaction is validated and committed

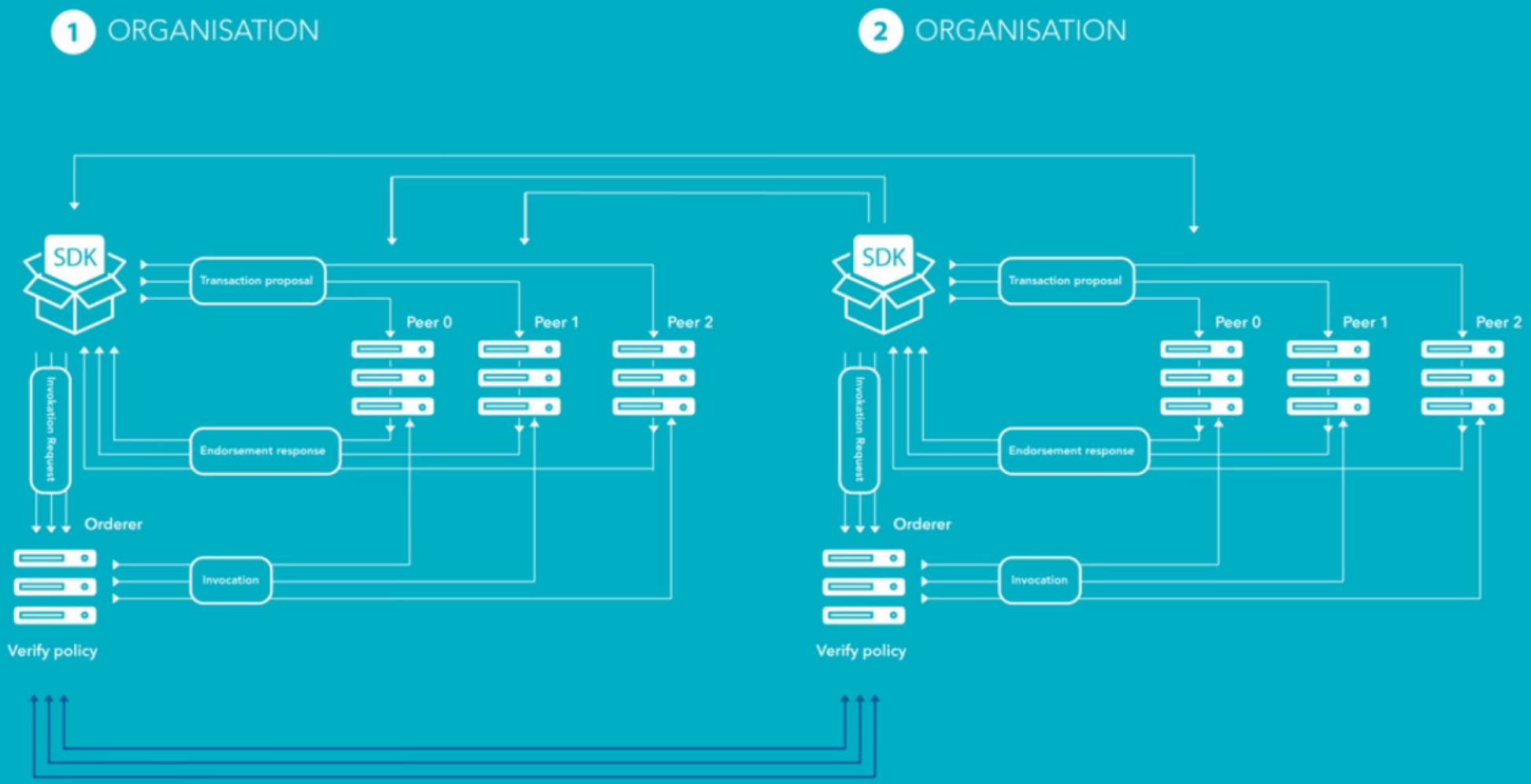


https://hyperledger-fabric.readthedocs.io/en/release-1.2/txflow.html

Ledger updated



https://hyperledger-fabric.readthedocs.io/en/release-1.2/txflow.html


1 ORGANISATION
2 ORGANISATION
SDK
Transaction proposal
Peer 0
Peer 1
Peer 2
Invokation Request
Endorsement response
Orderer
Invocation
Verify policy
SDK
Transaction proposal
Peer 0
Peer 1
Peer 2
Invokation Request
Endorsement response
Orderer
Invocation
Verify policy

Hyperledger

Live example…