CODETE

Introduction to Distributed Ledger Technology for Enterprise

Choosing the best blockchain implementation for your project

Karol Przystalski

8.11.2019

About me

Overview

2015 – obtained a Ph.D. in Computer Science @ Polish Science Academy and Jagiellonian University

2010 until now – CTO @ Codete

2007 - 2009 – Software Engineer @ IBM

Recent research papers

Multispectral skin patterns analysis using fractal methods, K. Przystalski and M. J.Ogorzalek. Expert Systems with Applications, 2017

`https://www.sciencedirect.com/science/article/pii/S0957417417304803`

Contact

karol@codete.com

0048 608508372

Environment

https://hub.docker.com/r/kprzystalski/dlt-ehtereum
https://hub.docker.com/r/kprzystalski/dlt-corda

https:
//github.com/kprzystalski/oreilly-intro-to-dlt

https://github.com/kprzystalski/
oreilly-intro-to-dlt/README.md

Agenda

1. Introduction
2. Ethereum
3. Hyperledger Fabric
4. Corda
5. Summary
6. Machine learning and distributed ledger technology

Introduction

Buzzwords, buzzwords and more buzzwords

Blockchain became a buzzword a few years ago. Like deep/machine learning or data science, each buzzword is often used by startup to show the innovative approach. Be aware the of many scam projects that are sold as blockchain-based. You can find some examples: `https://deadcoins.com/`.

Blockchain is still not a mature technology and the real-world use cases are limited. In many cases blockchain is used even a simpler, well-known solution can be used.

Questions I should answer . . .

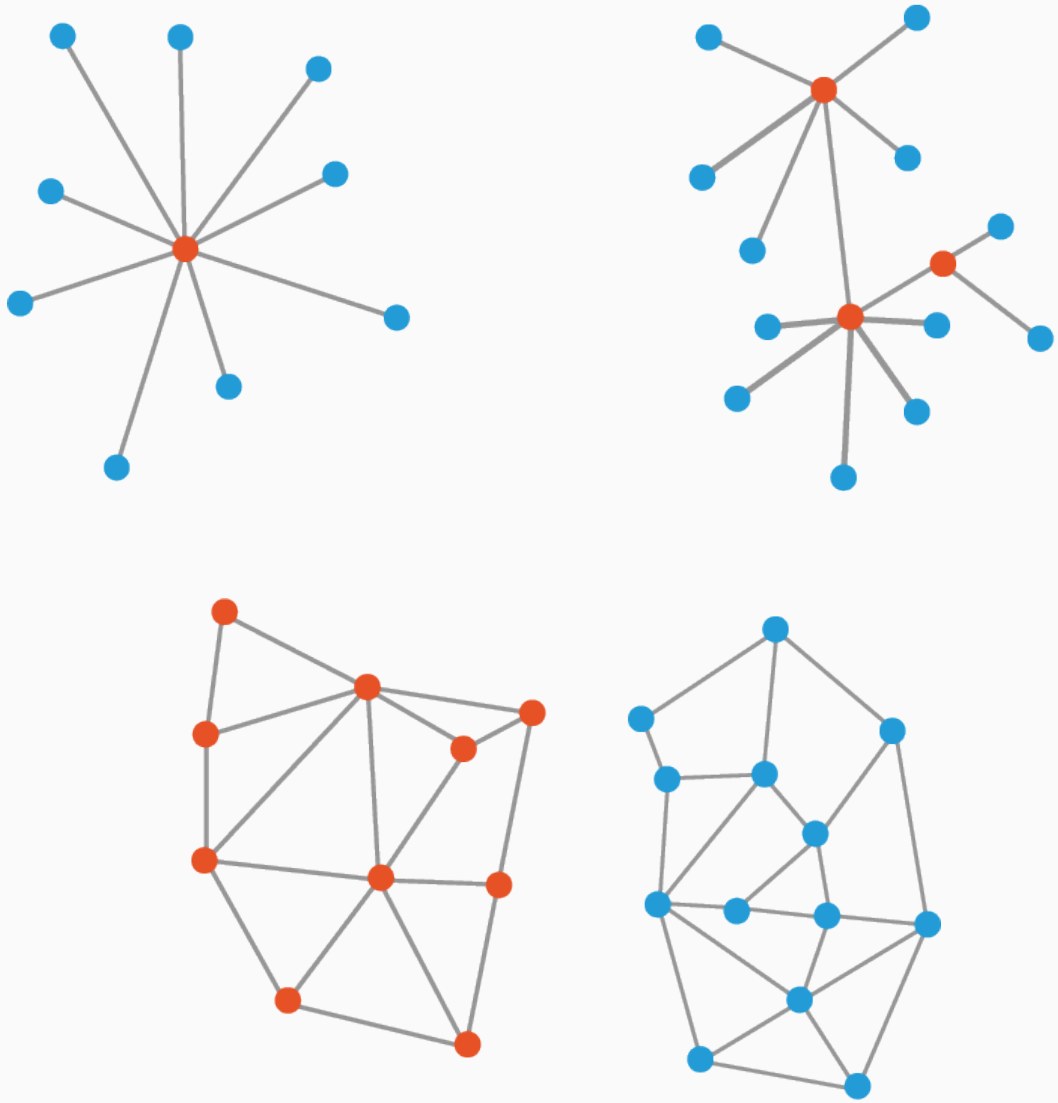
Before using blockchain you should answer yourself some questions:

1. Do I need a solution that does not allow to edit any added element/object (immutability)?
2. Should my app be distributed?
3. Is the security important?

If you answer **Yes** to all three question it still mean you should use Blockchain, but you can at least consider it as a potential solution for your application.

In many cases the Blockchain can be used to solve only a smaller part of our solution.

Approaches



What is a distributed ledger?

Distributed Ledger Technology is a term where nodes are used for calculation the agreements. The data is shared and synchronized between nodes.

There are two most popular implementation of DLT: Blockchain and IOTA.

Terminology

Based on `https://openblockchain.readthedocs.io/en/latest/protocol-spec/#1-introduction_1` there are a few important terms related to DLT.

Transaction is a request to the blockchain to execute a function on the ledger. The function is implemented by a chaincode.

Transactor is an entity that issues transactions such as a client application.

Ledger is a sequence of cryptographically linked blocks, containing transactions and current world state.

World State is the collection of variables containing the results of executed transactions.

Chaincode is an application-level code (a.k.a. smart contract) stored on the ledger as a part of a transaction. Chaincode runs transactions that may modify the world state.

Terminology

Validating Peer is a computer node on the network responsible for running consensus, validating transactions, and maintaining the ledger.

Non-validating Peer is a computer node on the network which functions as a proxy connecting transactors to the neighboring validating peers. A non-validating peer doesnt execute transactions but does verify them. It also hosts the event stream server and the REST service.

Permissioned Ledger is a blockchain network where each entity or node is required to be a member of the network. Anonymous nodes are not allowed to connect.

Terminology

Privacy is required by the chain transactors to conceal their identities on the network. While members of the network may examine the transactions, the transactions cant be linked to the transactor without special privilege.

Confidentiality is the ability to render the transaction content inaccessible to anyone other than the stakeholders of the transaction.

Auditability of the blockchain is required, as business usage of blockchain needs to comply with regulations to make it easy for regulators to investigate transaction records.

Taxonomy

We can divide the distributed ledgers depend on:

- Access – pubic (permissionedless), private (permissioned), hybrid (consortium)
- Chain – linked list, DAG, list of lists
- Consensus method - PoW, PoS, ...
- Crypto method used - SHA256, ...

Depending on the use case we should choose the right solution.

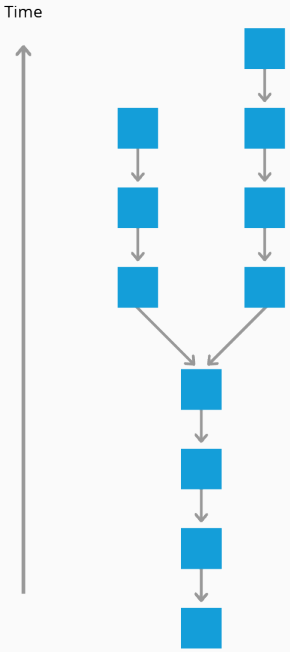
DLT solutions

Most DLT solutions are open source. The most popular are:

- Bitcoin – `https://github.com/bitcoin/bitcoin`,
- Ethereum – `https://www.ethereum.org/`,
- Hyperledger projects – `https://www.hyperledger.org/`,
- Neo – `https://neo.org/`,
- Corda – `https://www.corda.net/`,
- Lisk – `https://lisk.io/`,
- EOS – `https://eos.io/`,
- Hedera – `https://www.hedera.com/`.

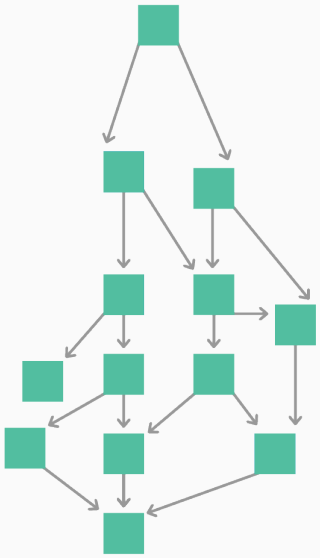
Different implementations of DLT

Blockchain



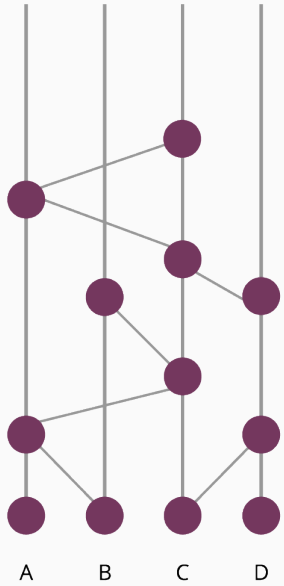
Linked list

Tangle



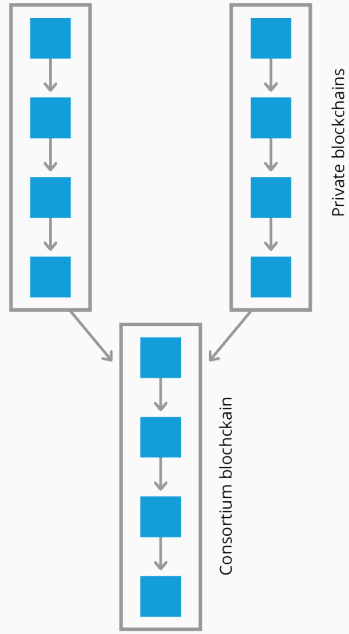
Directed acyclic graph

Hashgraph



Directed acyclic graph

Sidechain



List of Linked lists

Blockchain characteristics

Each blockchain consist of six layers.

Layers	Components
Data layer	Data block, timestamp, merkel tree, etc.
Network layer	Peer2Peer network
Consensus layer	PoW, PoS, ...
Service layer	Hyperledger, ...
Application layer	Cryptocurrency, healthcare

Table 1: Based on [4]

Blockchain evolution

Blockchain 1.0 was used for Bitcoin and was limited to:

- public only,
- simple transactions,
- proof of work as the consensus method.

In Blockchain 2.0, the current used one, the biggest change that was introduced are smart contracts. It is enhanced to use different consensus methods. Other features:

- private, public or hybrid,
- you can combine blockchains,
- different consensus methods available.

One of the most important major changes that will be introduced in Blockchain 3.0 is confidential computing. It allows smart contracts to access data outside of the blockchain in a secure way.

Security in DLT – crypto

There are two use cases in Blockchain:

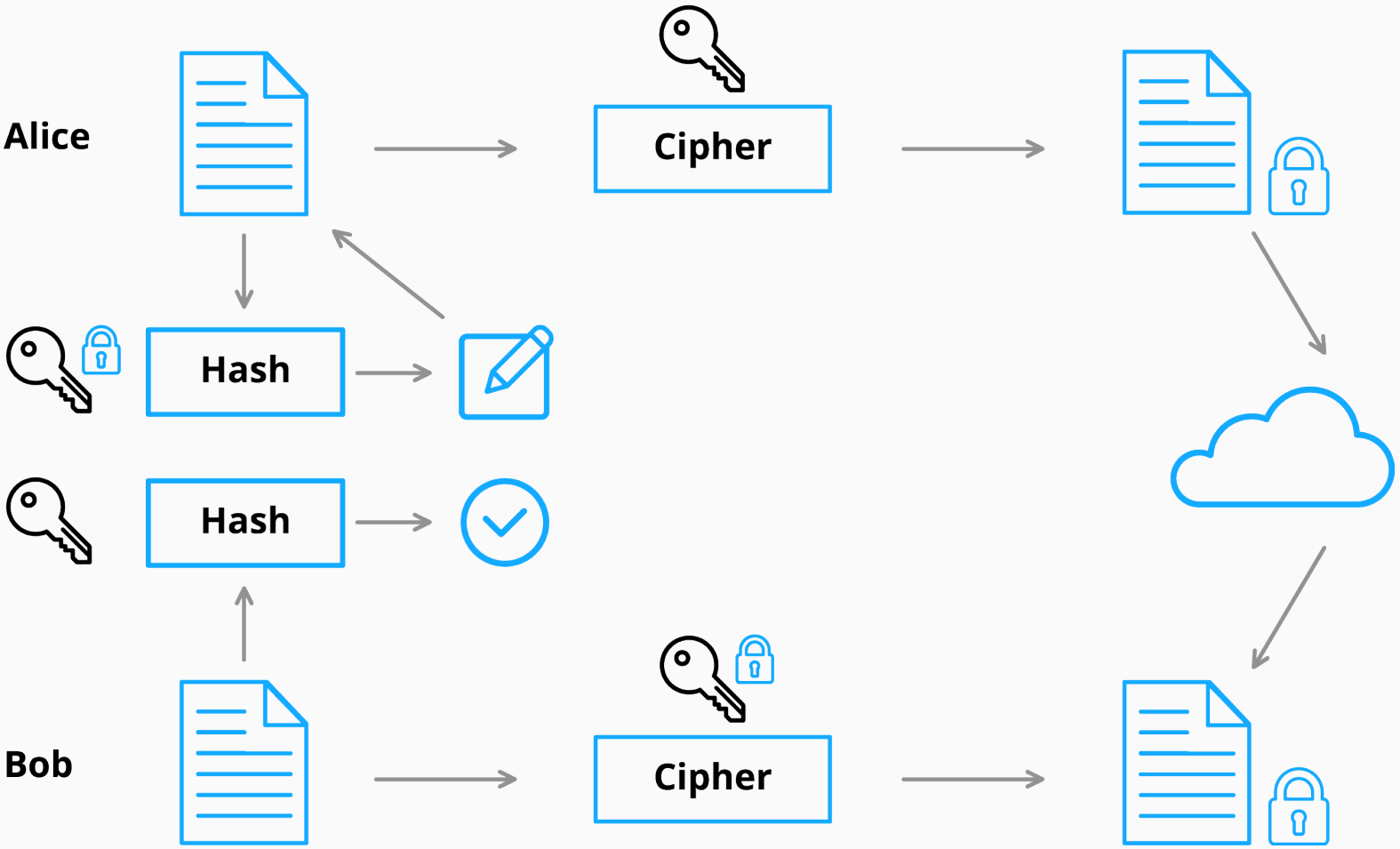
- data encryption,
- signature,
- peers communication encryption.

The methods of encryption are asymmetric. The default methods used by each solution are:

- Bitcoin – SHA256 for hashing and ECDSA for signature,
- Ethereum – KECCAK256 for hashing and ECDSA for signature,
- Corda – support several algorithms – `https://docs.corda.net/cipher-suites.html` ,
- Fabric – SHA3 SHAKE256 for hashing and ECDSA for signature by default.

Blind signature is used sign by other nodes.

Security in DLT – crypto



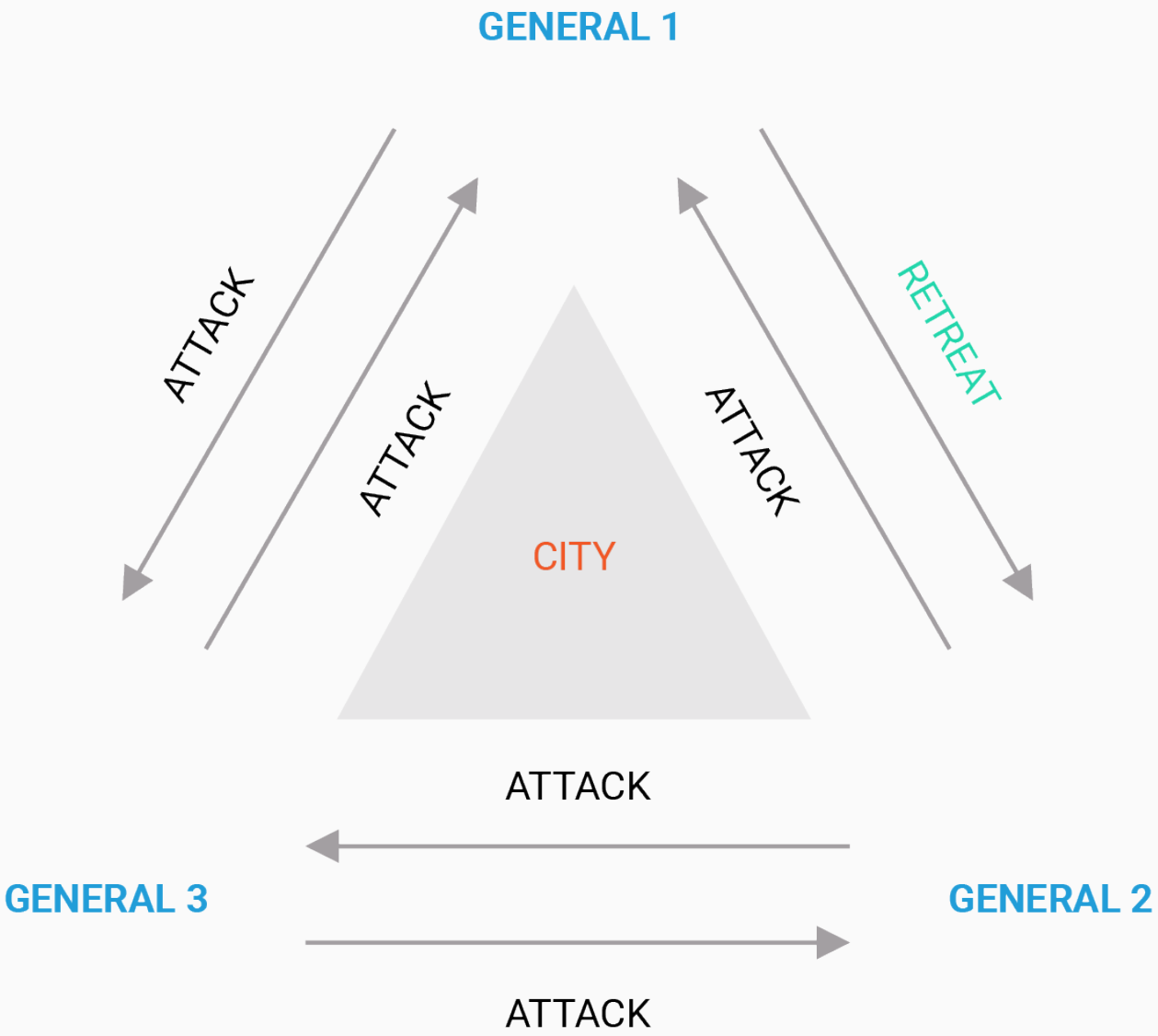
Security in DLT – attacks

Blockchain is secure by design, but as everything, blockchain is vulnerable to attacks. Depending on the layer, we can carry out different attack.

Layers	**Attack**
Network layer	DoS, Sybil or Eclipse attack
Consensus layer	Transaction delay attack, 51% attack
Service layer	Re-entrancy attack

Apart from the above attacks, we have a bunch of infrastructure attacks, not just on the blockchain itself, but attacks that affect blockchain.

Consensus – Byzantine generals problem



Consensus algorithms

There are many consensus methods, just to mention most popular:

- Proof of Work,
- Proof of Stake,
- Proof of Burn (PoB),
- Proof of Activity (PoA),
- Proof of Elapsed Time (PoET),
- Federated Byzantine Agreement (FBA),
- Deposit Based Consensus (DBC),
- Practical Byzantine Fault Tolerance (PBFT),

Proof of Work vs. Proof of Stake

Proof of work:

- to add a theeh miners need to solve a punk that takes computation,
- to add a malicious floor, you need more power than 51% of the whole network,
- the fastest miner to solve the puzzle get a reward.

Proof of stake:

- the blow creator is chosen by users' stake,
- to add malicious block you need to have 51%. of network stake,
- no rewards.

Consensus algorithms

Proof of Activity is a combination of proof of work and proof of stake. PoA use PoW part until new block is found. Next we switch to PoS. It takes less energy as Pow.

Proof of Burn base on the amount of coins that the user burns to validate a transaction. It's also expensive as PoW, but it consumes assets only.

Proof of Elapsed Time is used by Hyperledger Sawtooth. It uses Intel's SGX units.

Federated Byzantium Agreement is used by stellar. In FBA nodes can choose whom they trust. It's called as *quorum slice*.

Consensus algorithms

PBFT is a consensus method where each transaction is signed by a week. When enough signatures are reached then this transaction is considered to be valid. This method is fast and efficient, but all participants need to agree on the same list of participants. The agreement is set by a central authority. It's used by Hyperledger.

Deposit Based Consensus is where nodes need to place a deposit before adding blocks. There is a chain selection, rule called GHOST where at base a- history. If GHOST consider a transaction as invalid the user loss the deposit.

Consensus algorithms overview

	Power	Tokenization	Rate	Difficulty	Scalability
PoW	Computational	Yes	Low	Hashing Functions	Yes
PoS	Virtual-currency	Yes	High	Voting	Yes
PoET	TEE	No	Medium	Voting	Yes
PBFT	Closed Negotiation	No	Low	Agreement	No
DBC	Deposit	No	High	Agreement	Yes
FBA	Node Consideration	No	High	Agreement	No
PoB	Burned Assets	No	Medium	Unspent trans.	Yes
PoA	Computational	Yes	Low	Follow the Satoshi	Yes

Based on [3]

Other consensus method

In the tangle, when the new transaction is registered it choose two other transactions for verification.

Nano build a separate blockchain for each user. A consensus protocol is not needed, because the user has the total ownership of the blockchain. It also has an impact on the performance. Each transaction between two users appends twice, on each user blockchain.

Storage

There are different storage methods used depending on the implementation we choose. We can distinguish between two major approaches:

- the information is a part of the block,
- we keep only the reference (index) to the information in the block

Blockchain demands to keep the whole chain by each node. In case of Ethereum it's now more than 145GB.

Blockchain as a Service (BaaS)

There are several BaaS providers. The most popular:

- Google STRATO – https://console.cloud.google.com/marketplace/details/blockapps-public-project/ba-strato?pli=1,
- Amazon Managed Blockchain (AMB) – https://aws.amazon.com/managed-blockchain/,
- Amazon Quantum Ledger Database QLDB – https://aws.amazon.com/qldb/,
- Infura – https://infura.io/,
- Stratis – https://stratisplatform.com/,
- Azure Blockchain Workbench – https://azure.microsoft.com/en-us/features/blockchain-workbench/.

Ethereum

Ethereum overview

Ethereum was found in 2015 and is a implementation fo Blockchain 2.0. Most important features:

- it's a open-source public blockchain platform,
- it uses PoS consensus method and Ether token as the cryptocurrency (ETH), where WEI is 10^{-18} of ETH,
- as it's public, you can check the transactions: `http://etherescan.io/`.

Ethereum community is one of the biggest in blockchain.

Clients

There are many clients available, but the most popular are:

- eth – `https://github.com/ethereum/webthree-umbrella`,
- geth – `https://github.com/ethereum/go-ethereum`,
- pyeth – `https://github.com/ethereum/pyethapp`.

There are more clients, in almost each of the most popular programming languages:

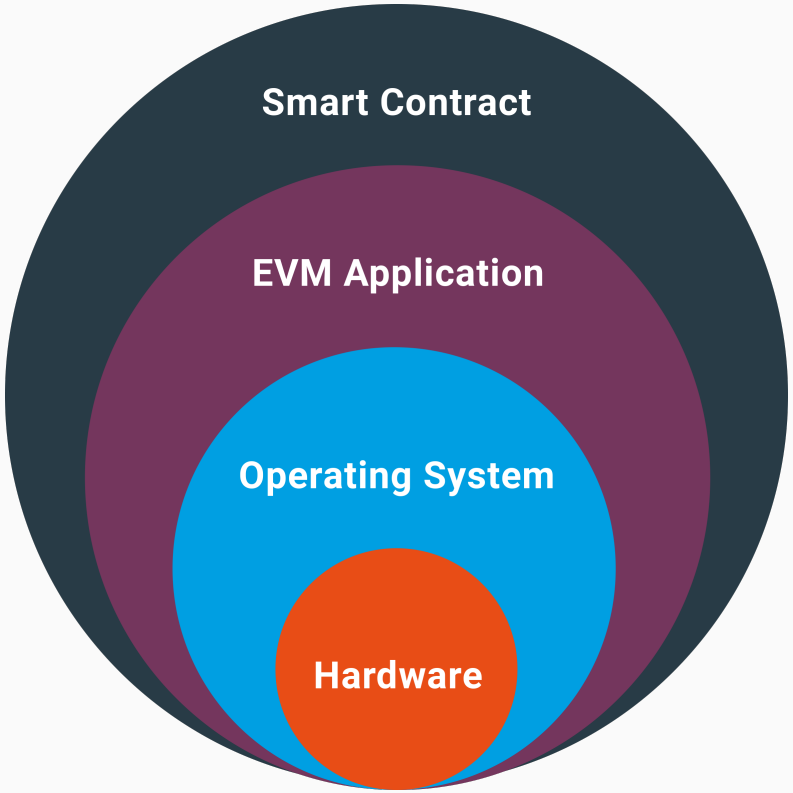
- parity – Rust,
- ethereumj – Java,
- ethereumH – Haskell,
- ruby-ethereum – Ruby.

Ethereum networks

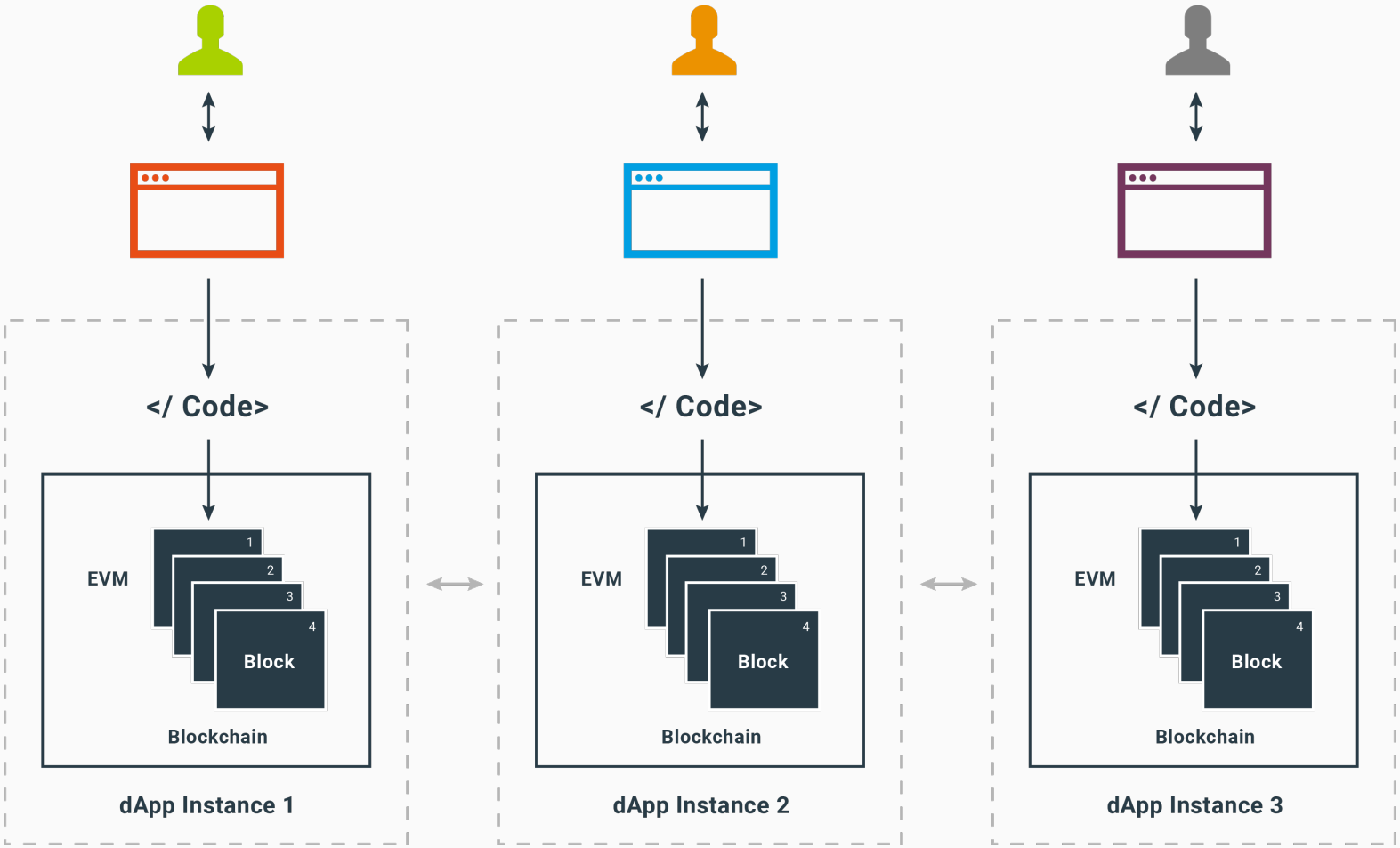
Ethereum provides several networks:

- Mainnet – main network,
- Ropsten – test network (PoW),
- Rinkeby – test network (PoA),
- Kovan – test network (PoA),
- Ethereum Classic Mainnet,
- Ethereum CLassic test network.

Environment



Distributed apps



Tools

Ethereum is developed using Go language. The tools are available in different languages. The most popular are implemented in JavaScript like:

- Drizzle,
- Truffle,
- Ganache.

You can check the tools at `https://www.trufflesuite.com/`.

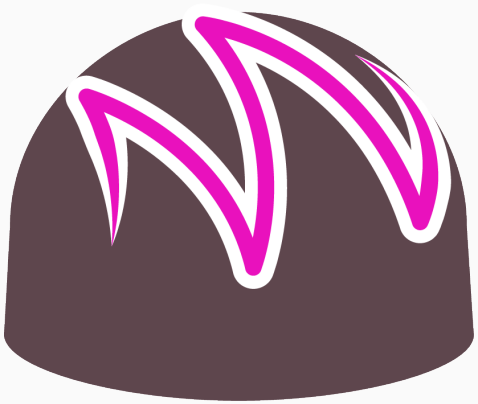
Smart contracts

There are three main languages used for smart contracts in Ethereum:

- Solidity. It is the main language for smart contracts in – `http://solidity.readthedocs.io`,
- Vyper. It is similar to Python. – `https://vyper.readthedocs.io`,
- Serpent – `https://github.com/ethereum/serpent/tree/develop/examples`.

Solidity smart contract demo

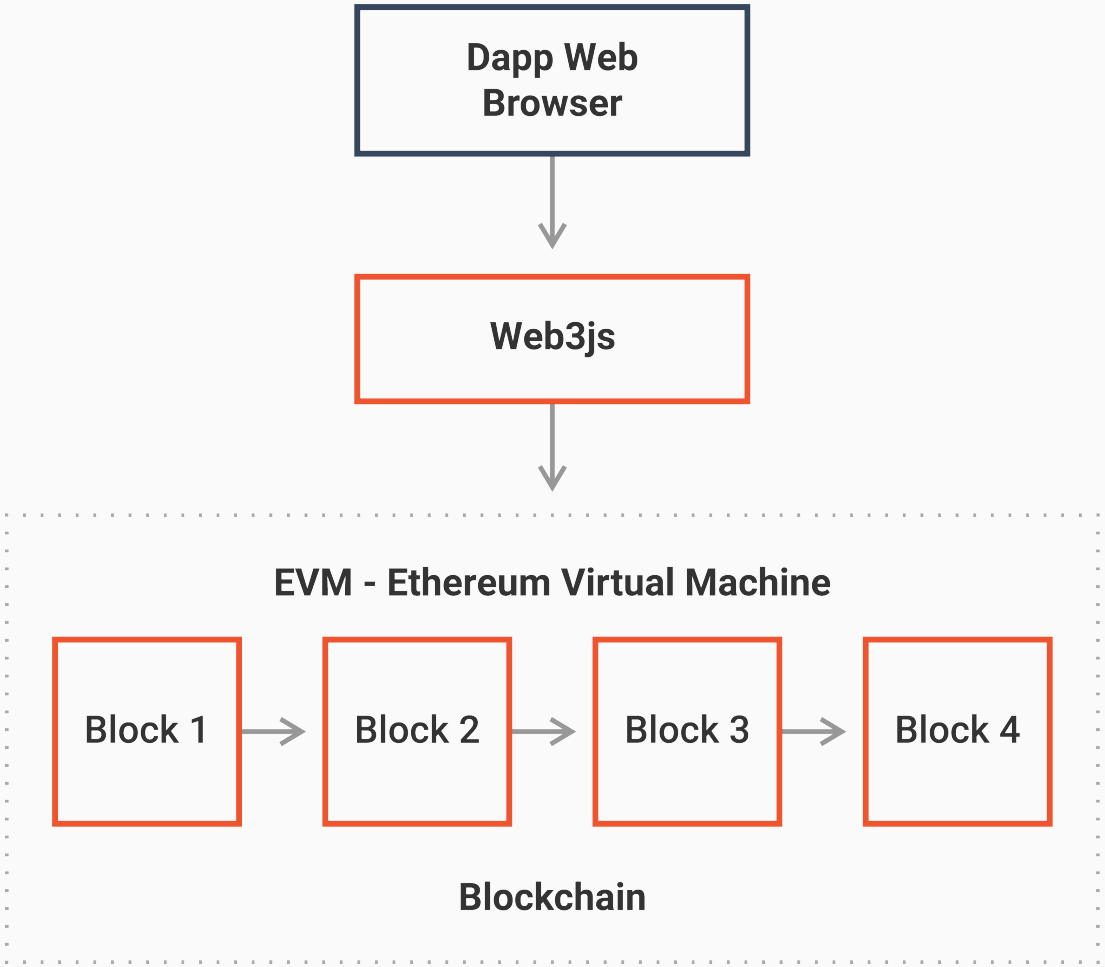
Drizzle



- generate the frontend for Ethereum distributed apps,
- integrates easily with React,
- stores the contract data on the frontend.

Drizzle demo

Voting distributed app



Pros and cons

You should use Ethereum when:

- you want to easily deploy your smart contracts and make it public,
- big community and support.

The limitations are:

- solidity,
- depend on the community.

Hyperledger Fabric

Hyperledger overview

Hyperledger is an open source blockchain implementation that was founded in 2015. Many companies are involved in this project. Some of the members are IBM, Intel, CISCO, SAP, Daimler, and American Express. Hyperledger project consists of several frameworks and tools for blockchain. The main features are:

- it's dedicated for enterprise,
- it's private,
- it's a frameworks, so you can easily change the default components,
- smart contract can be written in languages like Go, Kotlin or Java.

Hyperledger frameworks

HYPERLEDGER
BURROW

HYPERLEDGER
FABRIC

HYPERLEDGER
GRID

HYPERLEDGER
INDY

HYPERLEDGER
IROHA

HYPERLEDGER
SAWTOOTH

Hyperledger projects

Hyperledger Burrow is a client that interprets the smart contracts compatible with EVM.

Hyperledger Fabric is one of the most popular blockchain framework. It allows to choose the components you want and build a blockchain solution using docker containers.

Hyperledger Grid is dedicated for supply chain blockchain implementations. It provides data models, smart contracts and other components that are needed to build a supply chain based on blockchain.

Hyperledger projects

Hyperleger Indy is dedicated for digial identitiy projects. The project is mainly focused on privacy and privacy-preserving.

Hyperledger Iroha includes the BFT consensus method and is written in C++. Is allows to build your blockchain solution in Python, Java, JavaScript, C++ and for Android and iOS.

Hyperledger Sawtooth is a Intel supported solution where the PoET method is implemented. It does not have a central authority.

Hyperledger tools

Hyperledger tools

Hyperledger Caliper is a performance testing tool that provides tools to benchmark your blockchain solution. It provide such metrics like TPS or resource utilisation.

Hyperledger Cello provides a deployment model for blockchain solutions.

Hyperledger Composer is a set of tools to build blockchain networks and enables to create smart contracts and applications easier. You can deploy the application on Fabric.

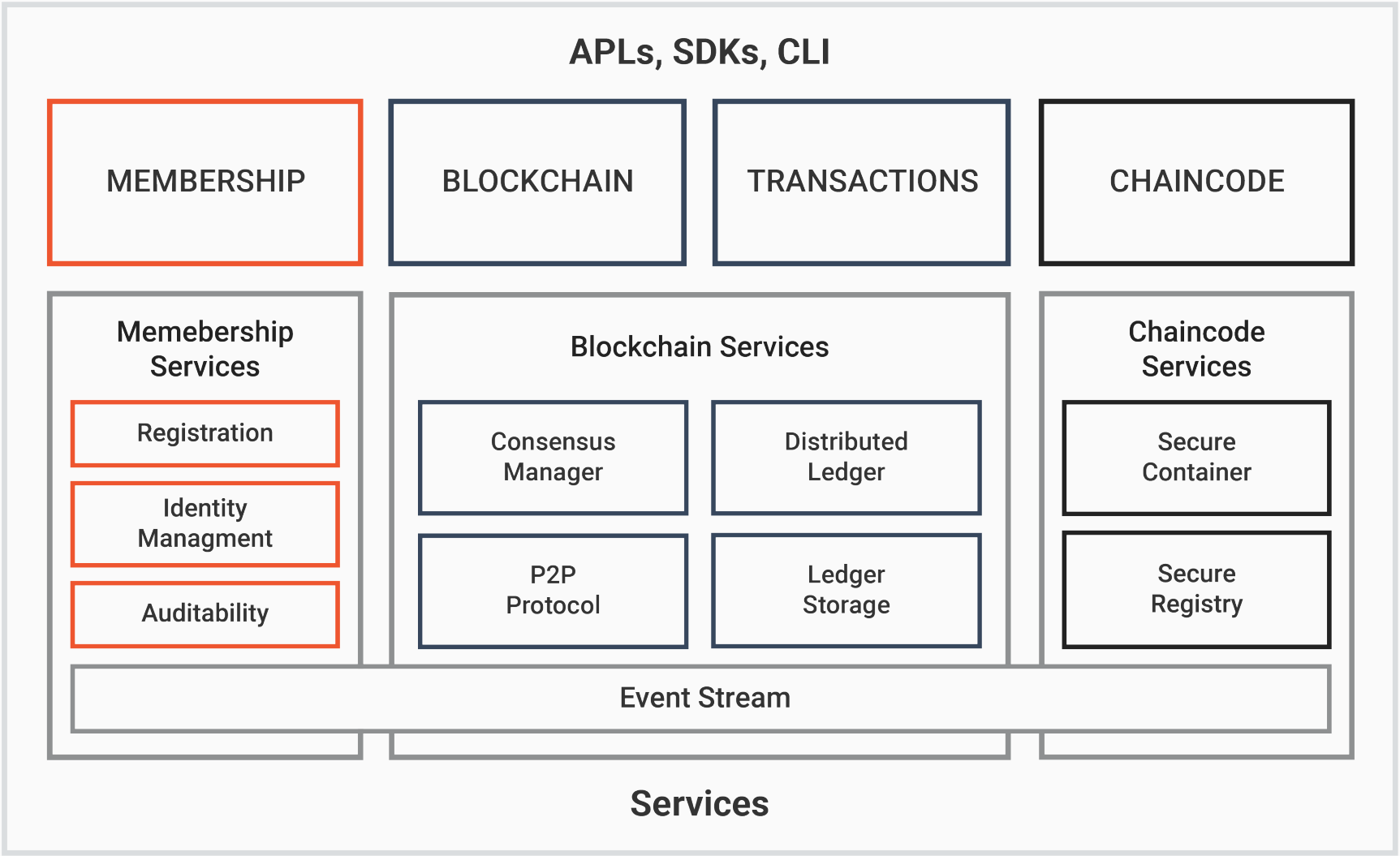
Hyperledger tools

Hyperledger Explorer is a web-based applciation that allows you to view, invoke, deploy, and many more in your blockchain application.

Hyperledger Quilt implements a interoperability protocol to move between ledger systems easier.

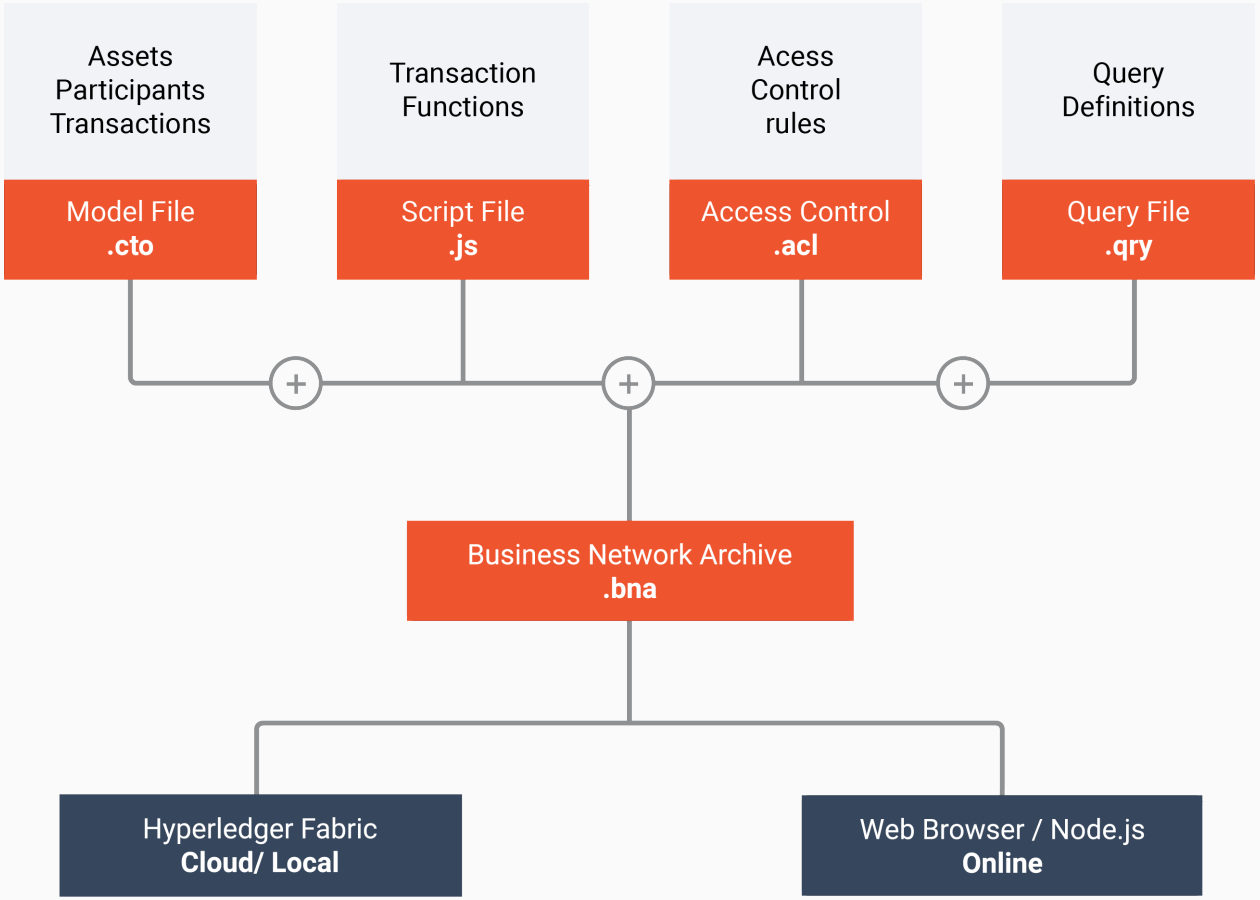
Hyperledger Ursa is a cryptographic library to avoid duplication and reduce the amount of cryptograhic work done.

Hyperledger Fabric architecture



Source: `https://openblockchain.readthedocs.io/en/latest/protocol-spec/#2-fabric_1`

Hyperledger Composer



Hyperledger Fabric Demo

Pros and cons

You should use Hyperledger Fabric when:

- you want to build a blockchain solution for internal use or want to control it,
- want to customize the solution by choosing the right components,
- easily to start with, because of Java that allow you to develop a solution faster from a HR point of view,
- scalability and performance,
- availability of channels.

The limitations are:

- permissioned and lack of transparency,
- complex architecture.

Corda

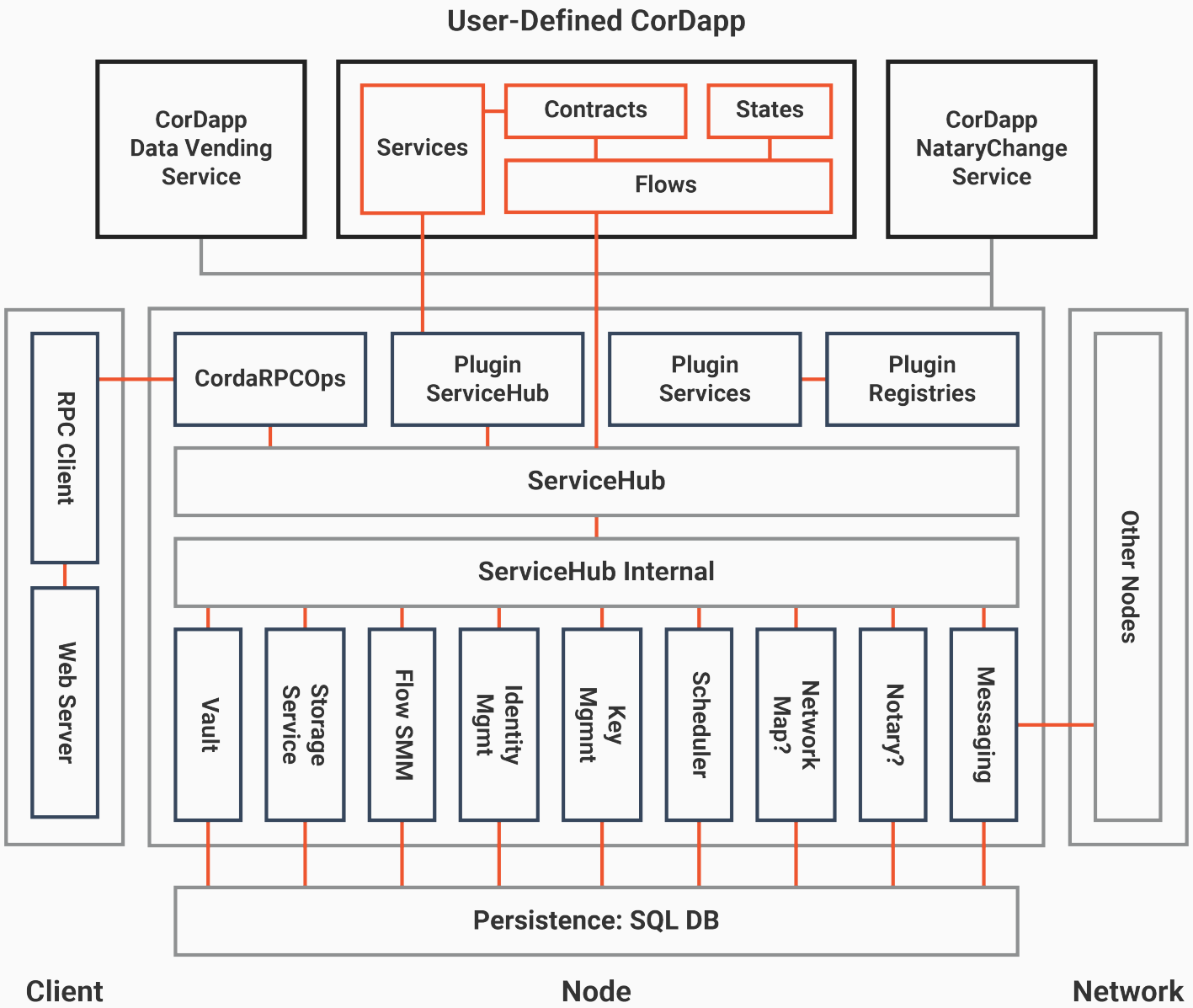
Introduction

R3 is a company that created Corda. There is an open-source and enterprise version of Corda.

Main feature of Corda:

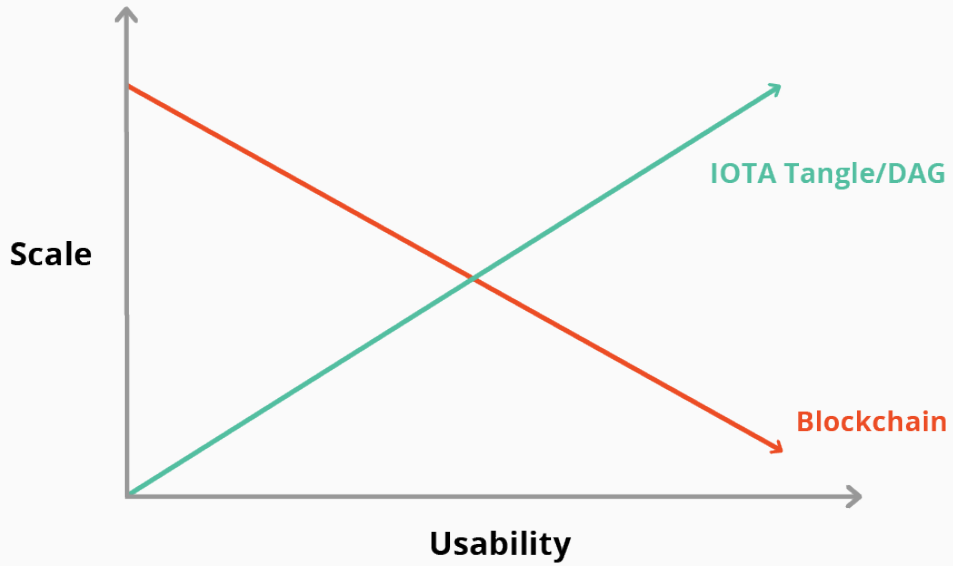
- no own cryptocurrency,
- is not a blockchain solution,
- agreement is done by participants that are related to the transaction only,
- no central authority.

Corda architecture

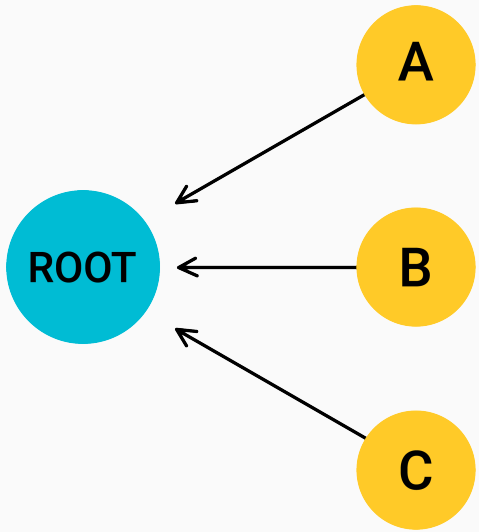


Blockchain vs. IOTA

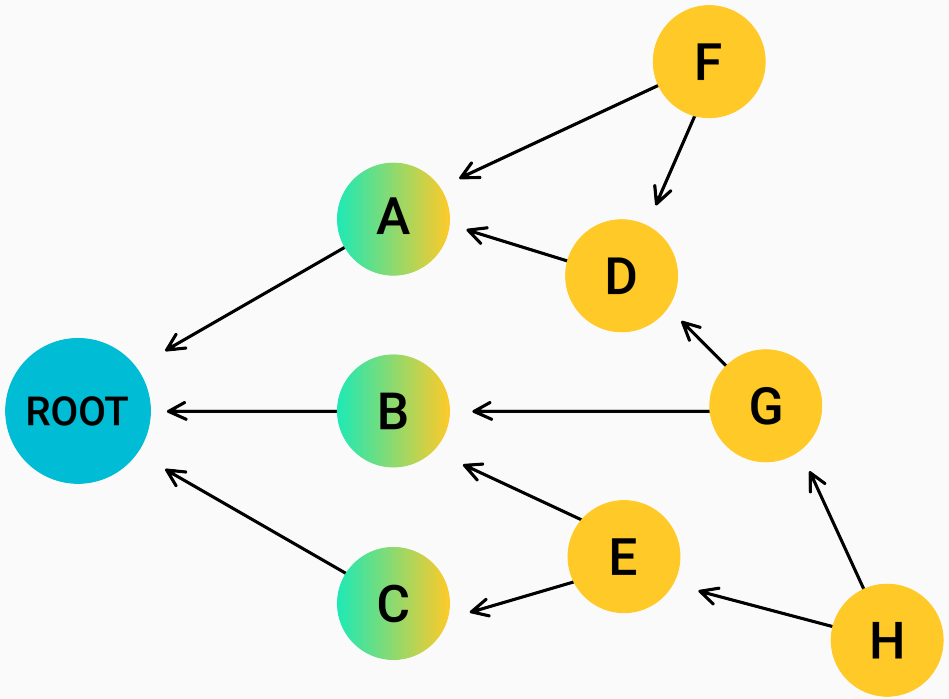
Blockchain and IOTA use a different consensus method. Based on the method it takes more or less time to add a new transaction to the chain. There are different benchmarks published, but in most cases IOTA-based solutions are faster than blockchain-based.



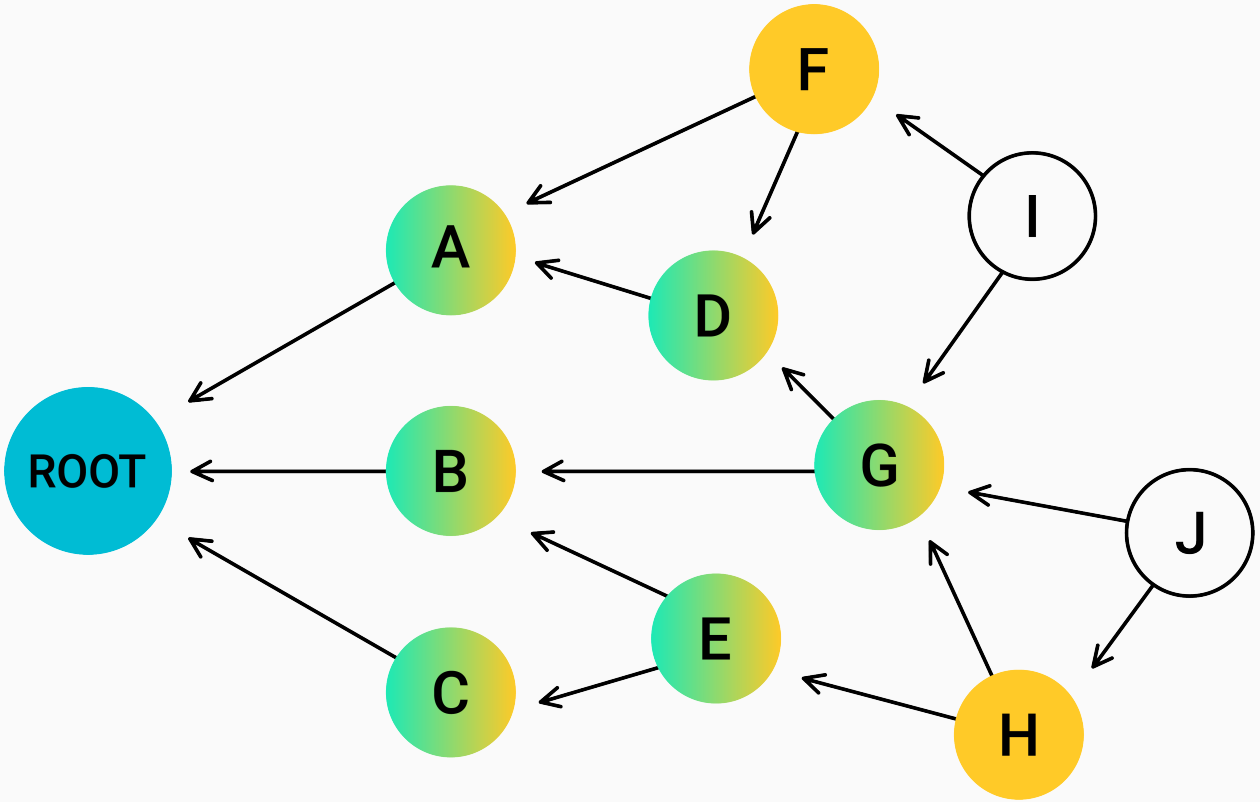
Consensus



Consensus



Consensus



Corda Demo

Corda and Hyperledger

Corda and Hyperledger are the most popular solutions used for enterprise. Both allows to build your smart contracts using different programming languages. Both are private and open source. Use same crypto algorithms. Corda and Hyperledger use different consensus methods – IOTA vs. PBFT. Use different currencies, and handle the contracts differently.

Pros and cons

You should use Hyperledger Fabric when:

- you want to build a blockchain solution for internal use or want to control it,
- want to customize the solution by choosing the right components,
- easily to start with, because of Java that allow you to develop a solution faster from a HR point of view,
- scalability and performance,
- availability of channels.

The limitations are:

- permissioned and lack of transparency,
- complex architecture.

Summary

What is the best solution?

There is no silver bullet here. Choosing the right DLT solution should be based on specific needs. You can answer several question to get the right solution.

Should it be public? **Yes** Ethereum **No** Corda or Fabric

Should I be able to choose the consensus method? **Yes** Fabric **No** Corda, Ethereum

Do I want to have a central authority? **Yes** Fabric **No** Corda, Ethereum

Does it really need to be blockchain? **Yes** Fabric, Ethereum **No** Corda

Machine learning and distributed ledger technology

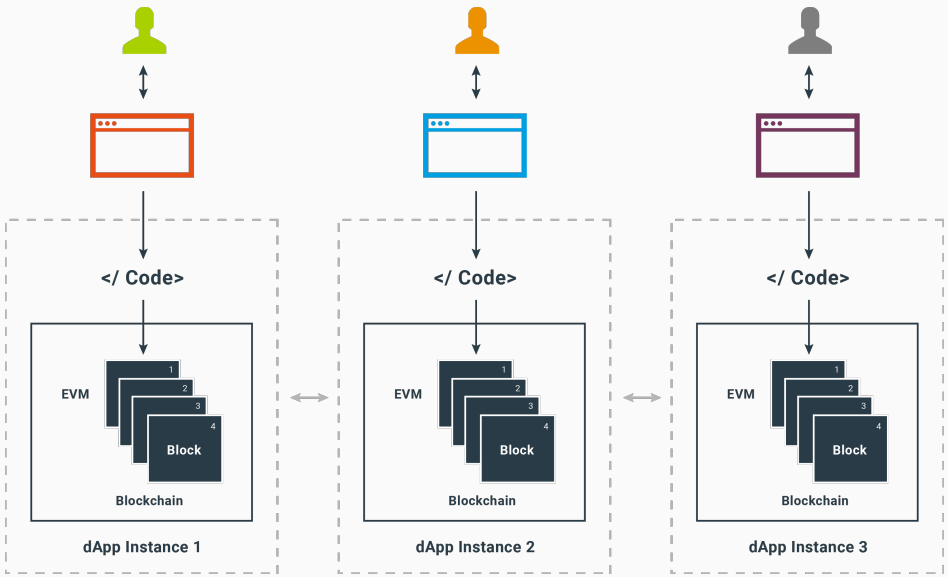
Security

Machine learning can be used to detect frauds, especially in solutions where the consensus methods can be broken.

Machine learning can be also used for discovering attacks on our blockchain. We could easily prevent it when recognized earlier.

Smart contracts

Technically, it's possible to include machine learning models into a smart contract. Even if it's possible, doesn't mean you should do that. It's known as decentralized/distributed machine learning.



Smart contracts for regulated solutions

Blockchain can be used as a network for gathering anonymous data from the users for the machine learning analysis.

It makes sense in healthtech, fintech or any other industry where the data is sensitive.

Blockchain as a part of the solution

The KODAKOne platform:

- monitors,
- protects copyright imagery online.

Wherever the platform detects an unlicensed picture usage we offer a friendly and pragmatic approach to help ensure the content creators get paid fairly for their work.

Proof of learning, Proof of AI

There are several consensus protocols where machine learning methods are used, like PoL [1] and PoAI [2]. The goal is to reduce the amount of time and energy by achieving the consensus.

Q&A session

References i

Felipe Bravo-Marquez, Steve Reeves, and Martin Ugarte.

Proof-of-learning: a blockchain consensus mechanism based on machine learning competitions.

01 2019.

Jianwen Chen, Kai Duan, Rumin Zhang, Liaoyuan Zeng, and Wenyi Wang.

An AI based super nodes selection algorithm in blockchain networks.

CoRR, abs/1808.00216, 2018.

J. Moubarak, E. Filiol, and M. Chamoun.

Comparative analysis of blockchain technologies and tor network: Two faces of the same reality?

2017 1st Cyber Security in Networking Conference (CSNet), 2017.

References ii

L. Yang.

The blockchain: State-of-the-art and research challenges.

Journal of Industrial Information Integration, page 10, 2019.