TECHNOLOGY

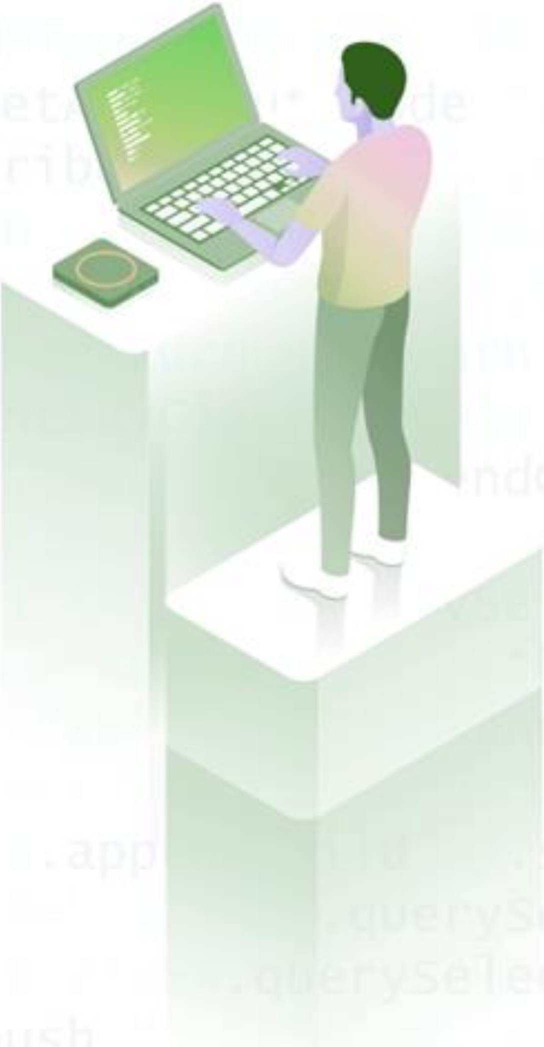
IIT KANPUR

Indian Institute of Technology, Kanpur

Professional Certification Program in Blockchain

Powered by simplilearn

TECHNOLOGY



Enterprise Blockchain

Powered by simplilearn

IIT KANPUR
Indian Institute of Technology, Kanpur

Learning Objectives

By the end of this lesson, you will be able to:

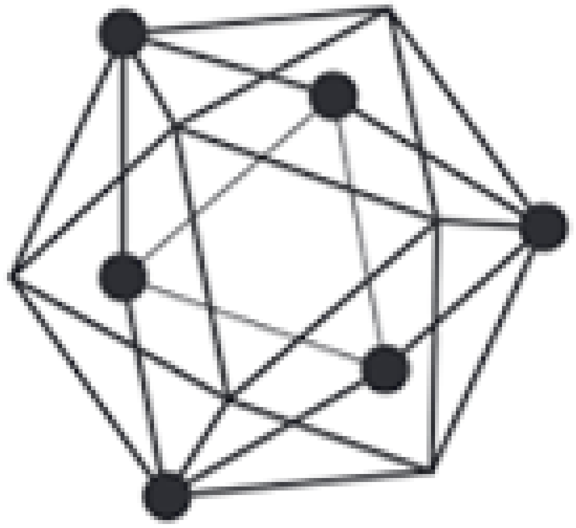
- Understand Enterprise Blockchain and its properties
- Identify Hyperledger Sawtooth, Iroha, Indy, Burrow, and Fabric
- List Hyperledger Fabric components, transaction cycle steps, and network types
- Analyze Corda and its network

TECHNOLOGY

Enterprise Blockchain

Enterprise Blockchain

Enterprise Blockchain is permissioned Blockchain that streamlines the business processes extensively, such as trail of supply chain goods, or solve global payments.

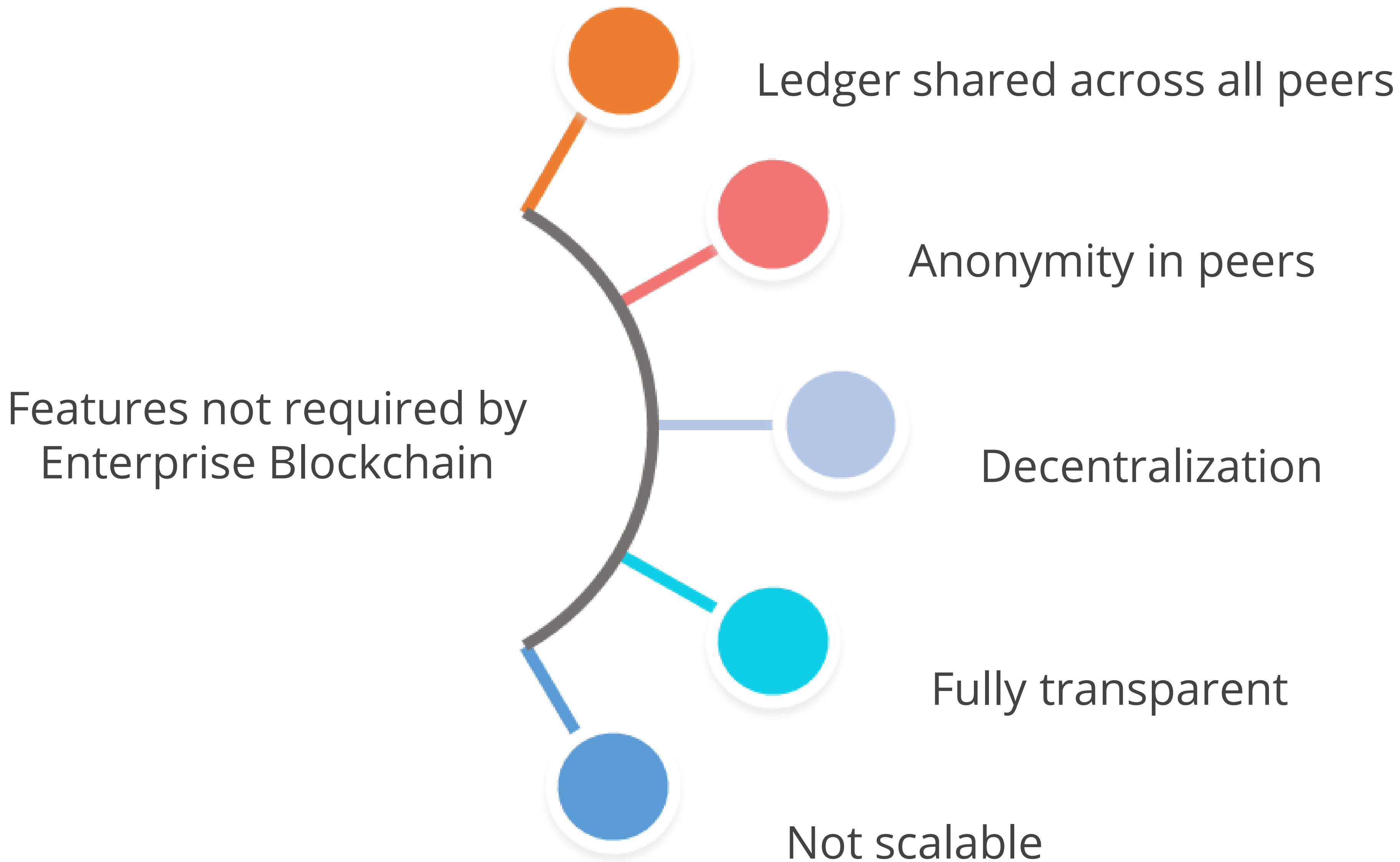


Hyperledger

Corda

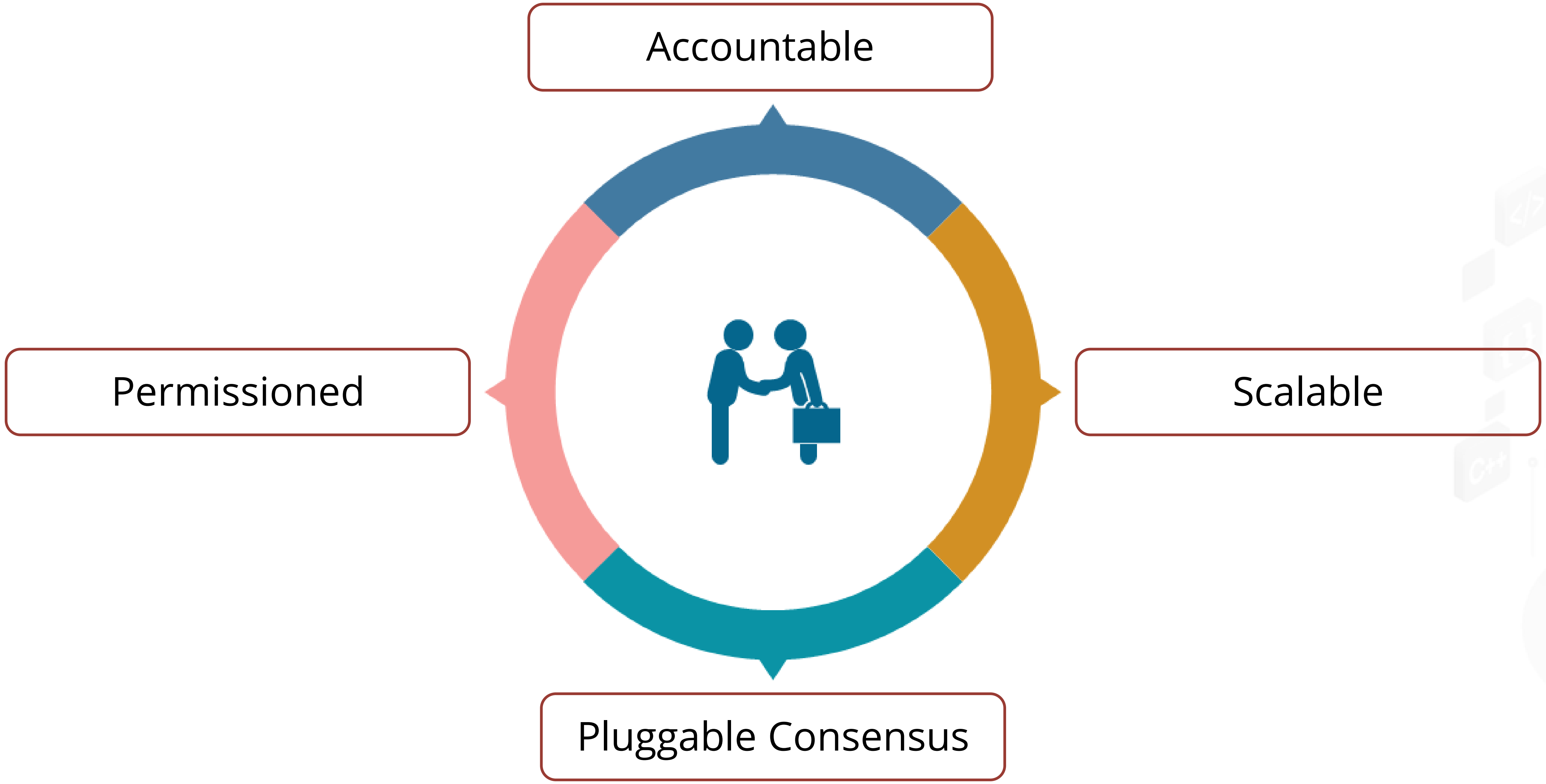
Enterprise Blockchain

Enterprise Blockchain is a permissioned Blockchain which is contrary to Public Blockchain.


Features not required by Enterprise Blockchain

- Ledger shared across all peers
- Anonymity in peers
- Decentralization
- Fully transparent
- Not scalable

Enterprise Blockchain Features



Accountable

Permissioned

Scalable

Pluggable Consensus

Hyperledger

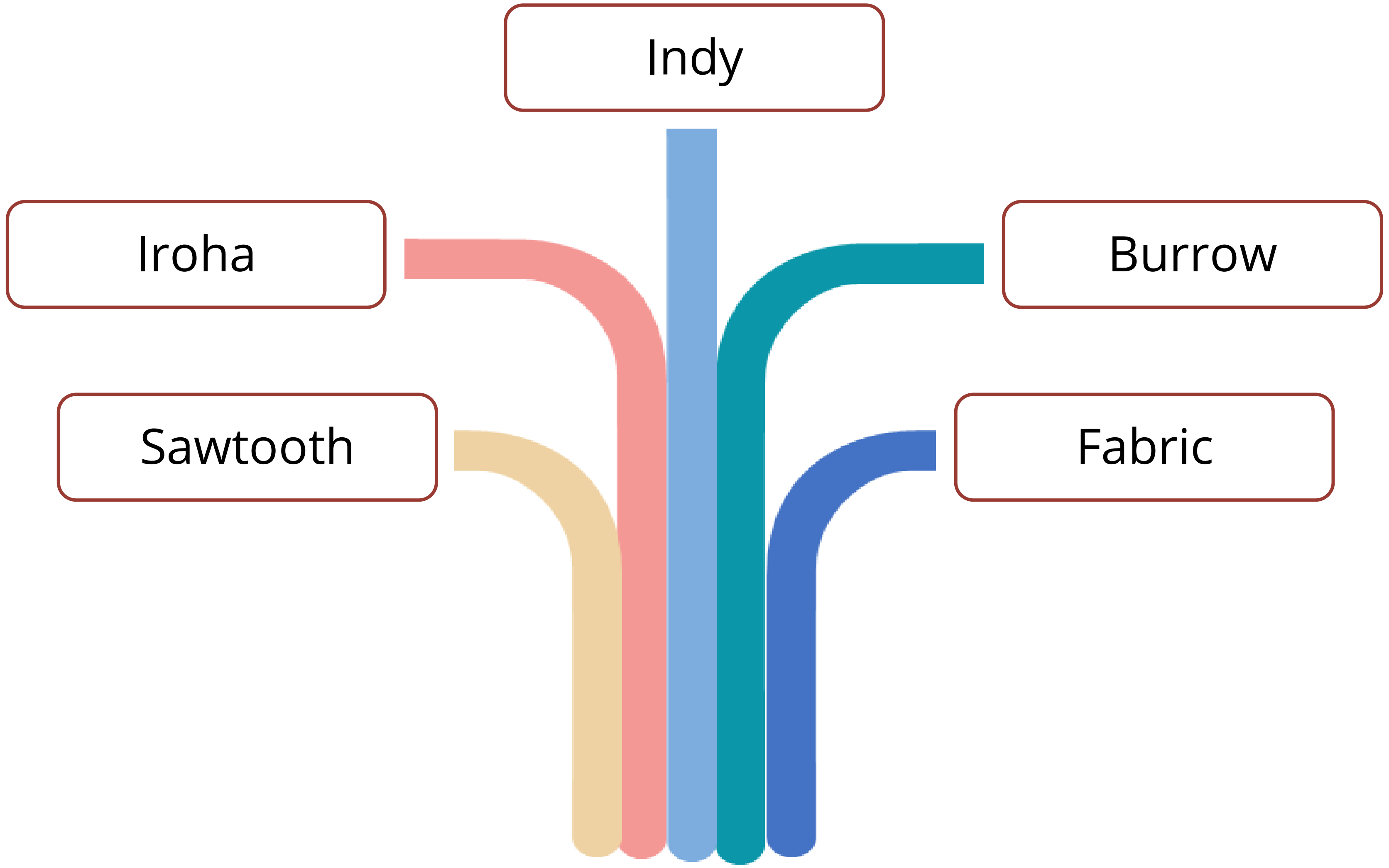
Introduction to Hyperledger

Hyperledger is a global enterprise blockchain project that provides the architecture, standards, guidelines, and resources needed to create open source blockchains and related applications.

Hyperledger

Hyperledger Umbrella Project

Hyperledger Umbrella Project comprises tools, services, and libraries to build Enterprise Blockchain applications.



Indy

Iroha

Burrow

Sawtooth

Fabric

TECHNOLOGY

Hyperledger Sawtooth

Introduction to Sawtooth

Hyperledger Sawtooth is an open source Blockchain framework to develop enterprise decentralized applications and networks. This simplifies process of development and deployment by isolating the core system from the application domain.

HYPERLEDGER
SAWTOOTH

Advantages of Sawtooth

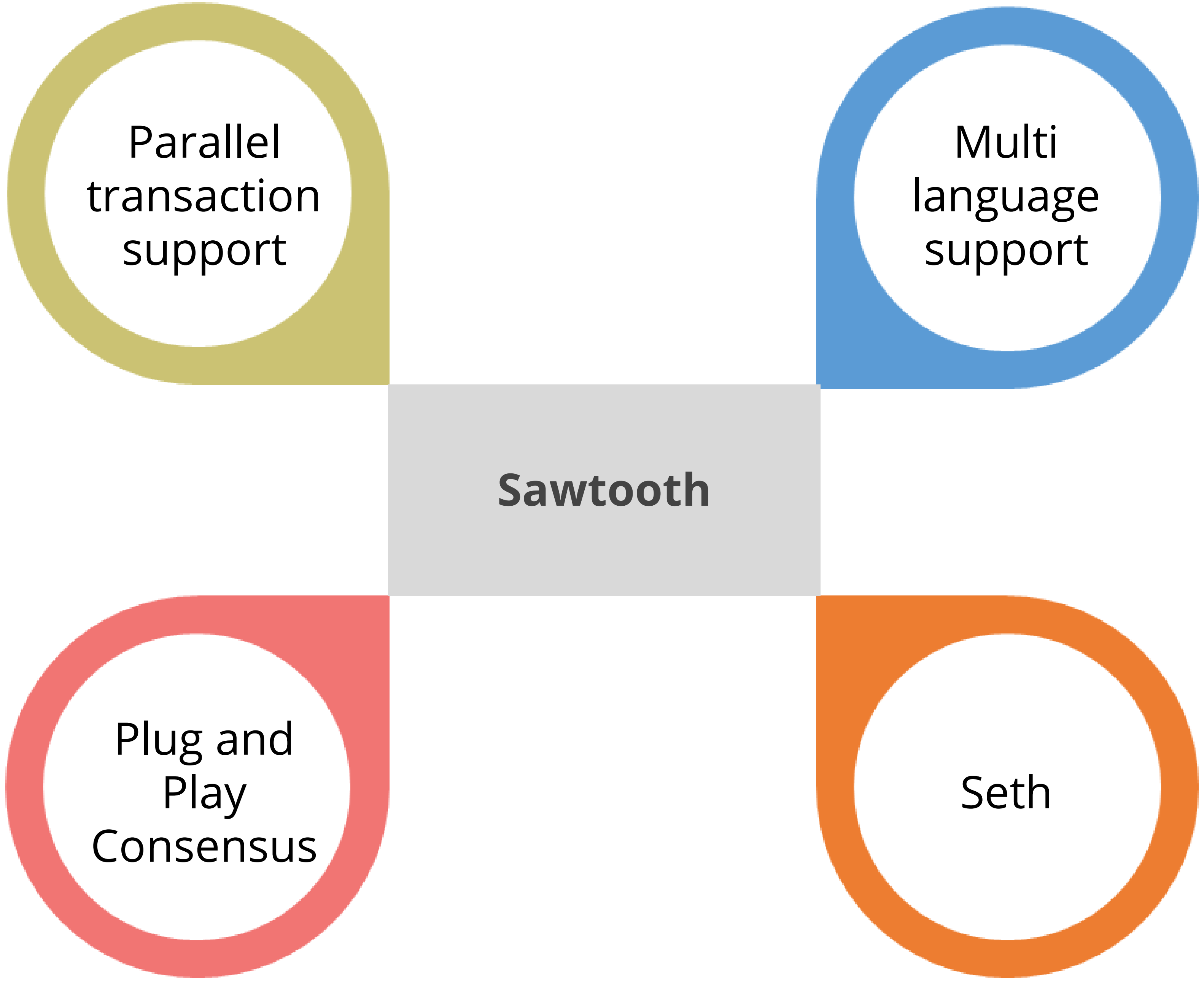
Is scalable

Is easily developed and deployed

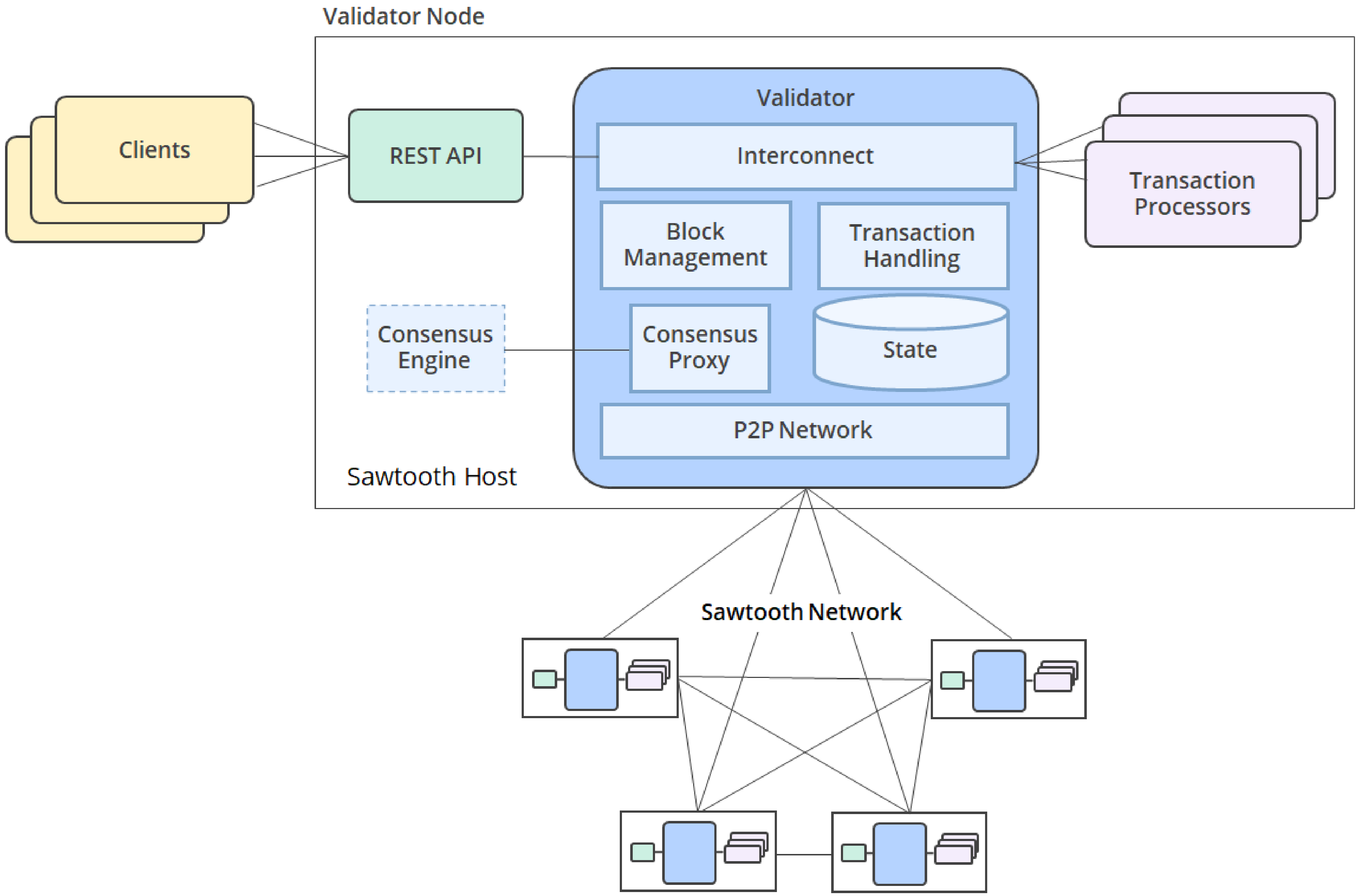
Maintains distributed nature of Blockchain

Keeps smart contract safe

Sawtooth Features



Sawtooth Architecture



Sawtooth architecture has five components:

1. Client
2. Rest API
3. Validator
4. Transaction Processor
5. Sawtooth Network

Sawtooth Architecture

Client

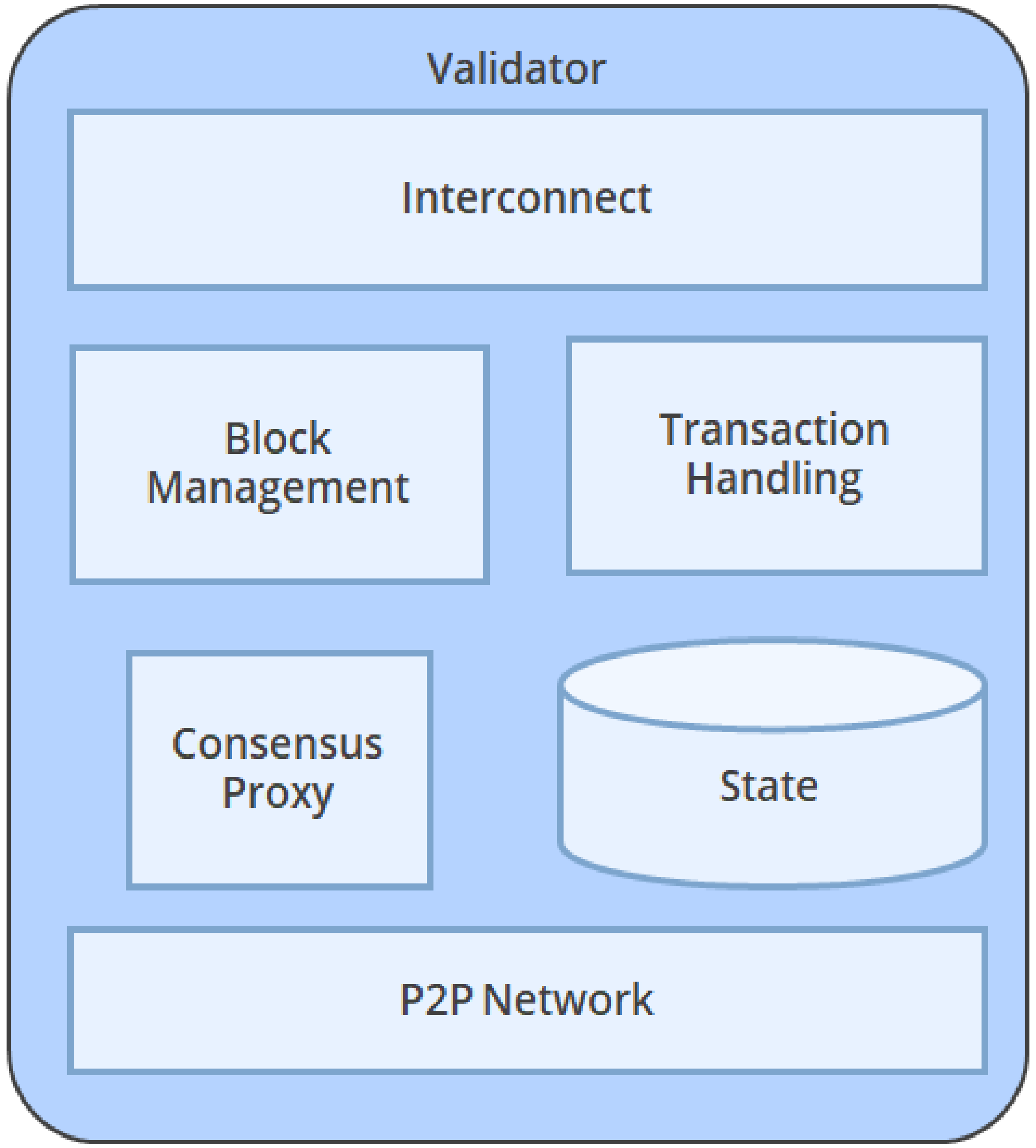
- A user or an application that makes the request to Sawtooth host
- Query the data from ledger
- Make a transaction to write to ledger
- Interact with Sawtooth host through Rest API call

Sawtooth Architecture

Rest API

- Intermediator between the Client and the Validator
- Medium for submitting transactions and reading blocks
- Performs different operations associated with Rest methods

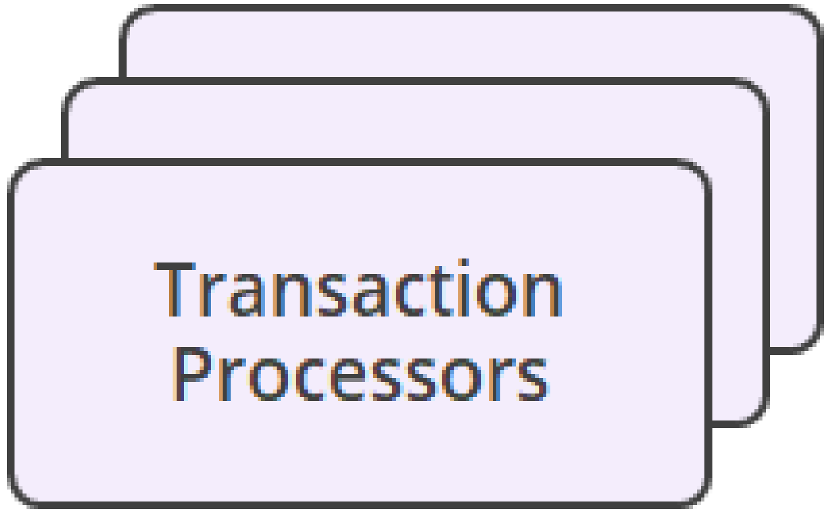
Sawtooth Architecture



Validator

Responsible for validating the incoming transactions and sending them to the transaction processor for further processing

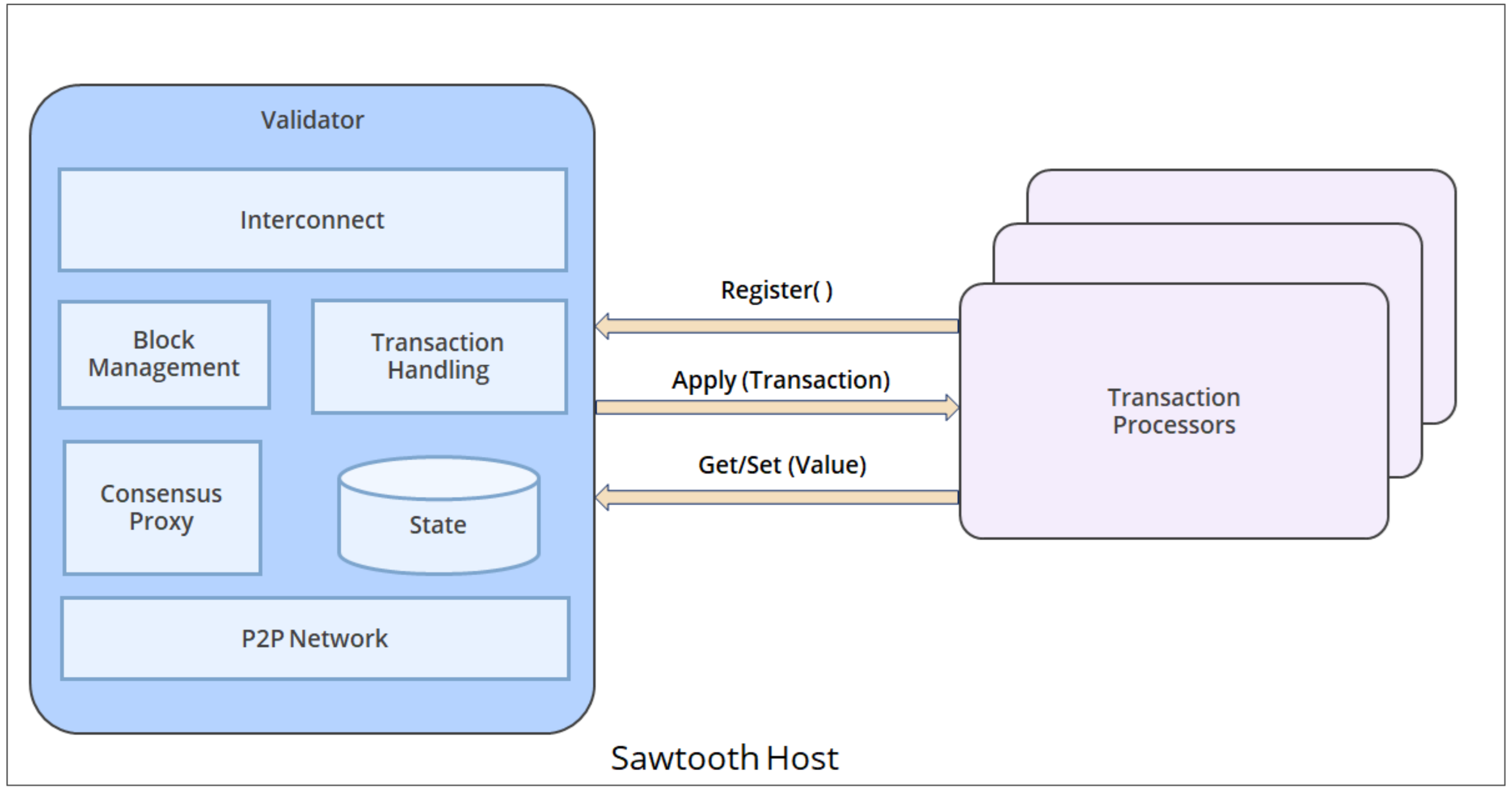
Sawtooth Architecture



Transaction Processors

- Compute the business logic that is equivalent to smart contract
- Constitute multiple transaction processors in one sawtooth host
- Example: Car ownership change

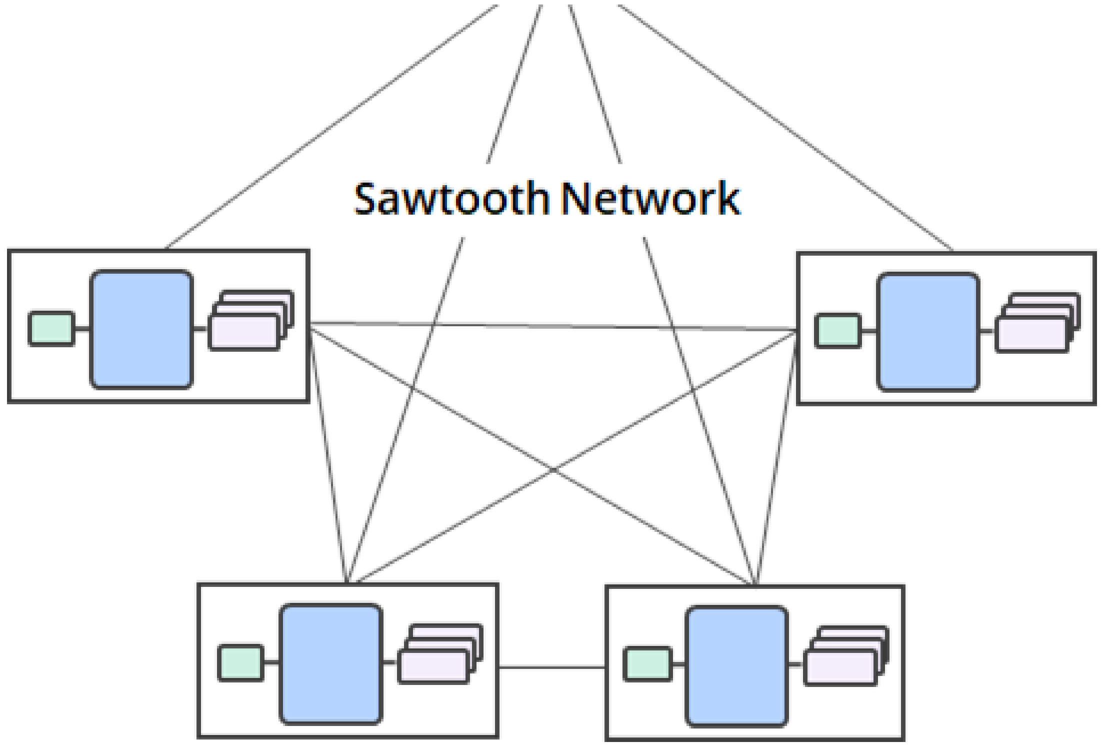
Sawtooth Architecture



APIs

- **Register():** Transaction processors get registered with validator
- **Apply(Transaction):** Dispatch the transaction details to the Transaction processor from validator
- **Get/Set(Value):** Read or write the latest key value pair in State Database

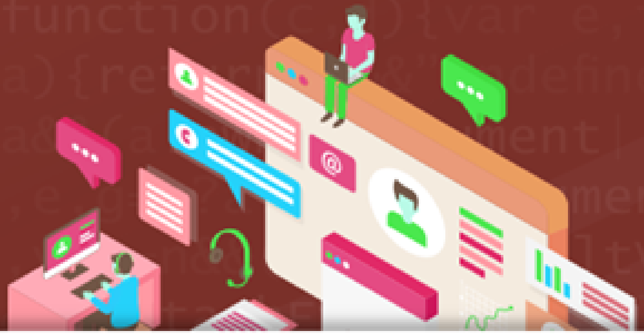
Sawtooth Architecture



Sawtooth Network

- Each host system is a Sawtooth node
- It operates one validator, a consensus engine, and a set of transaction processors

Set up Sawtooth Network and Create Basic Transaction



Problem Statement: You are given a task to set up Sawtooth network and run one sample.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to set up Sawtooth network and create basic transaction:

1. Upgrading the Docker Compose (optional)
2. Installing the Sawtooth network file
3. Creating the basic transaction in the Sawtooth network

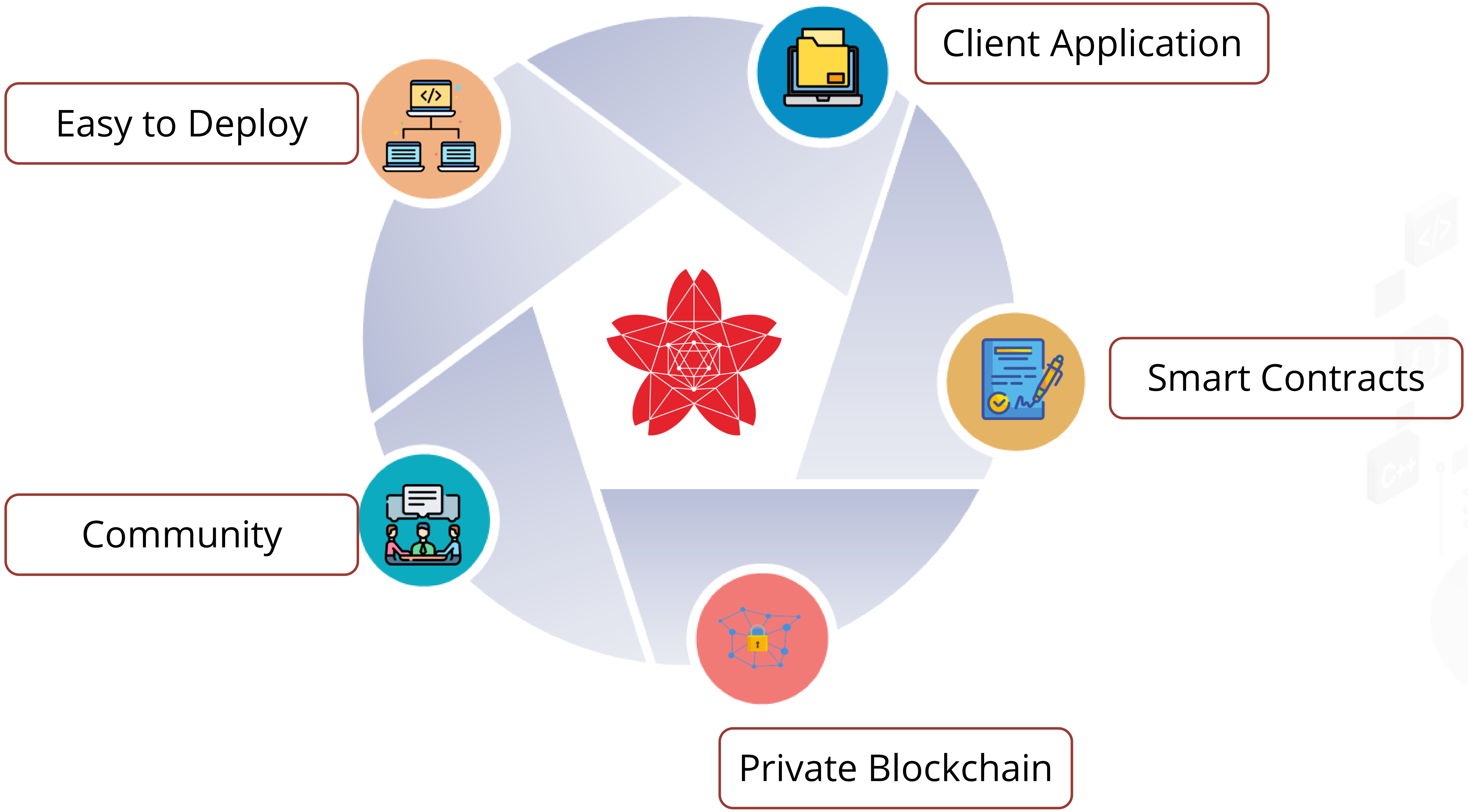
Hyperledger Iroha

Hyperledger Iroha

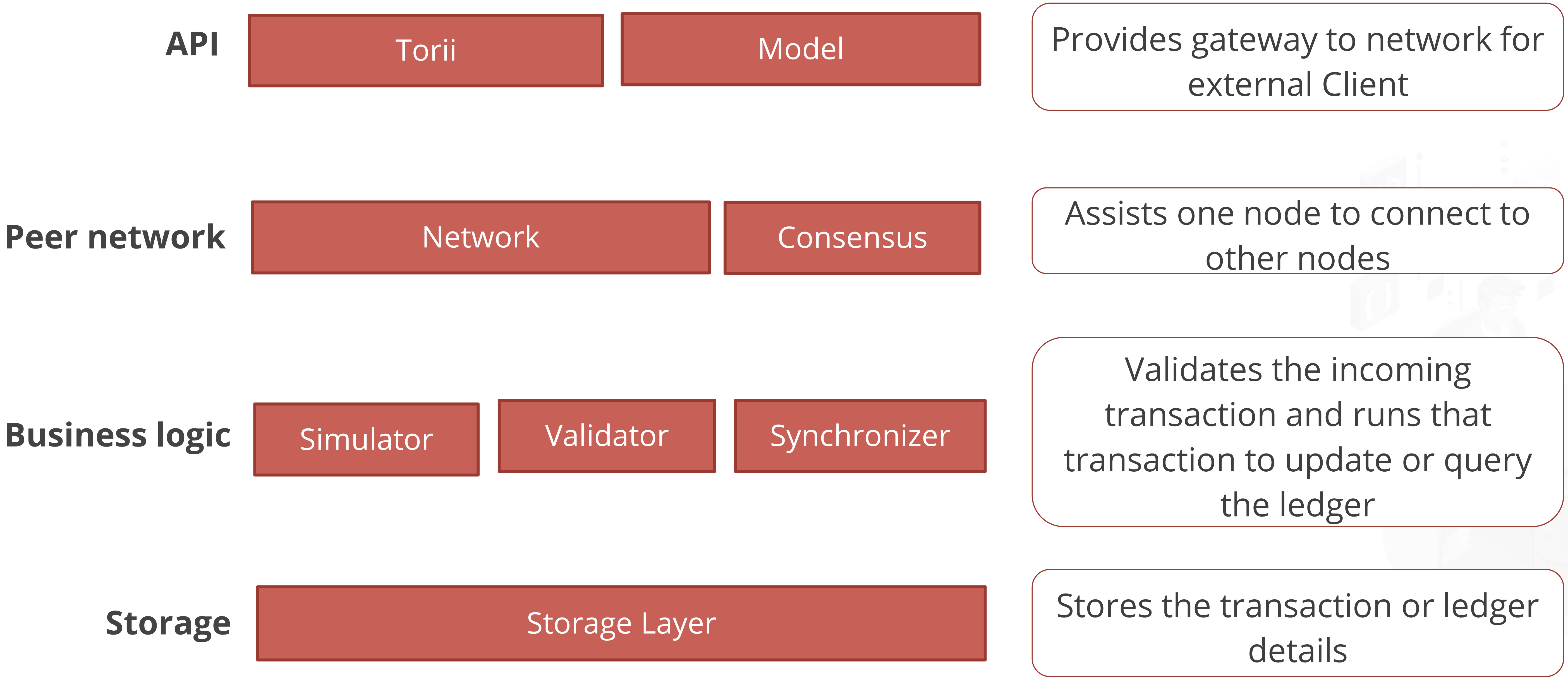
Hyperledger Iroha is a simple Blockchain platform that can be utilized to make trusted, secure, and fast applications that leverage the power of permission-based Blockchain with Byzantine fault-tolerant consensus.

Hyperledger Iroha

Hyperledger Iroha Features



Hyperledger Iroha Architecture

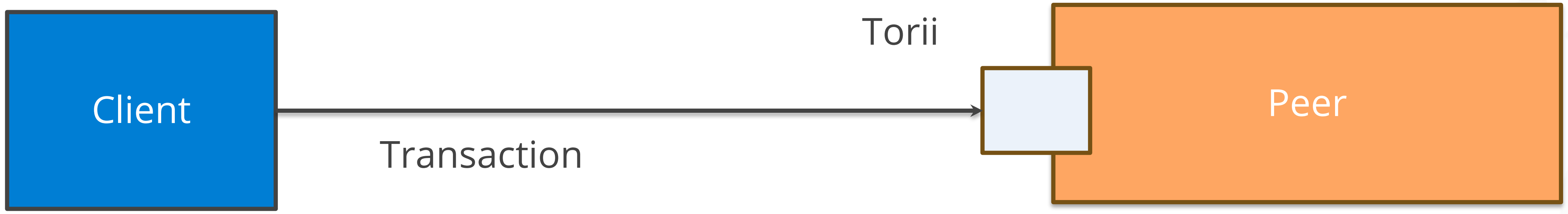


API	Torii	Model		Provides gateway to network for external Client
Peer network	Network	Consensus		Assists one node to connect to other nodes
Business logic	Simulator	Validator	Synchronizer	Validates the incoming transaction and runs that transaction to update or query the ledger
Storage	Storage Layer			Stores the transaction or ledger details

Transaction Flow

Step 1

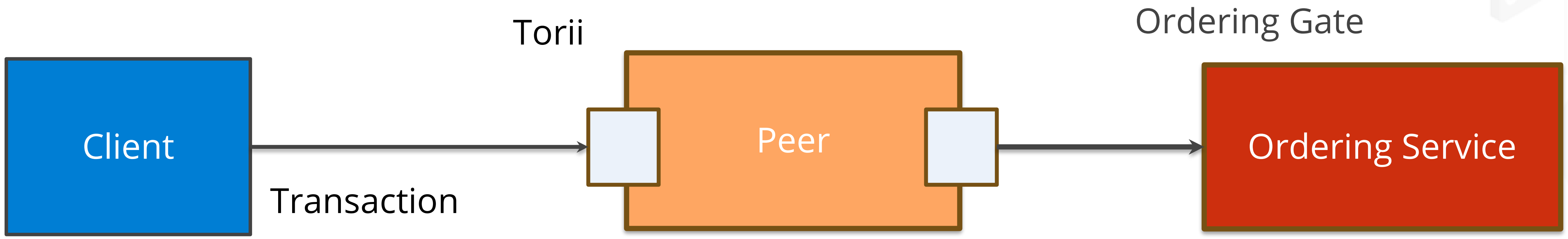
Client initiates the transaction and sends it to the peer through Torii gate



Transaction Flow

Step 2

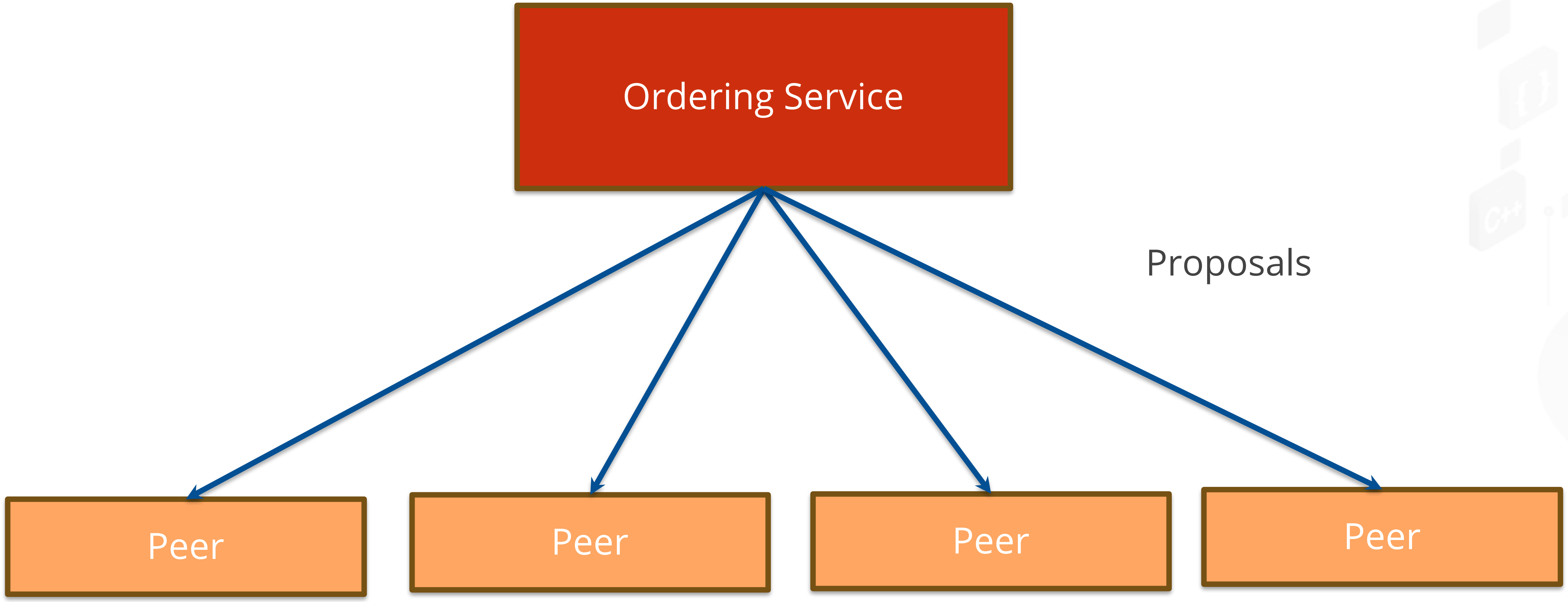
Peer sends the transaction to the Ordering Service through the Ordering Gate after performing stateless validation



Transaction Flow

Step 3

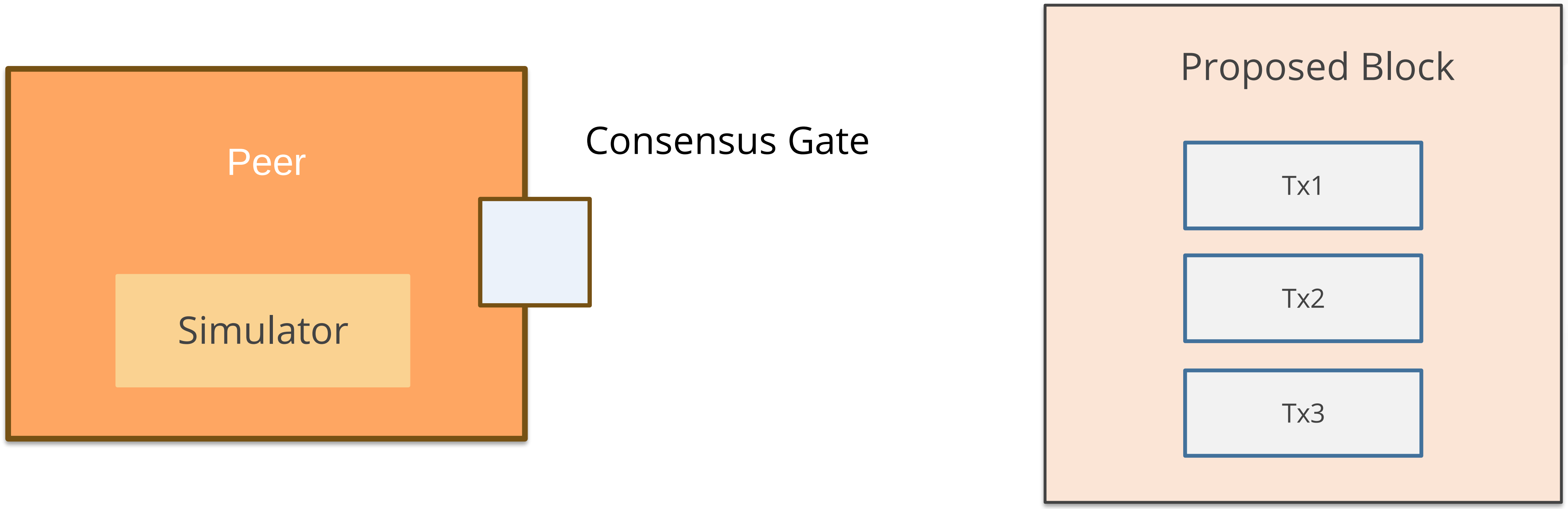
Ordering service collects the transactions and maintains a sequence of them. Transactions are then put into a block and they are sent to all peers as Proposals



Transaction Flow

Step 4

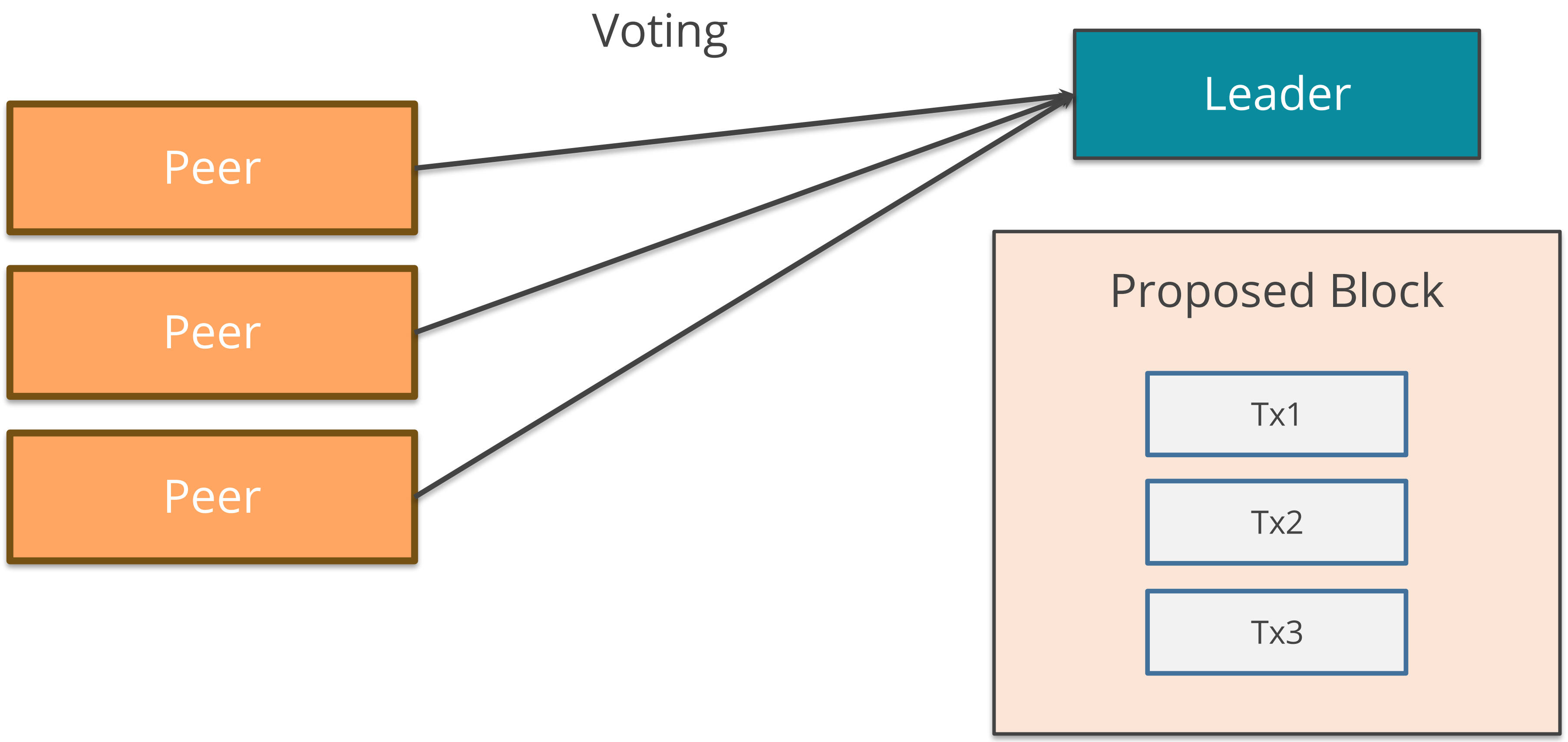
Proposed block reaches the peer, the peer runs the smart contract, and it simulates the transactions present in that block with the help of a simulator



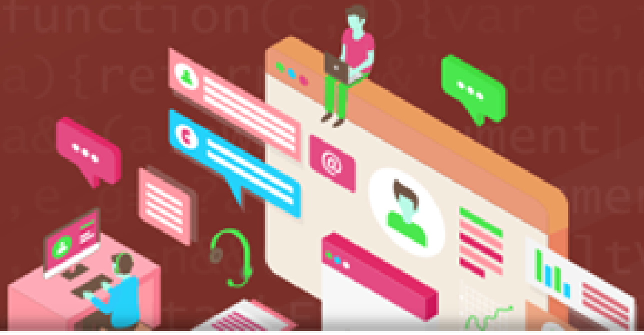
Transaction Flow

Step 5

All the peers choose a leader based on majority and then the leader confirms the block to be added to the ledger



Set up Iroha Network and Create Basic Transaction



Problem Statement: You are given a task to set up Iroha network and run one sample.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to set up Iroha network and create basic transaction:

1. Installing the Docker containers
2. Installing the Iroha network file
3. Creating the basic transaction in the Iroha network

Hyperledger Indy

Hyperledger Indy

Hyperledger Indy is a distributed ledger used to generate and store decentralized identities, allowing users to manage and control their digital identities. It stores pointers to identities rather than private data.

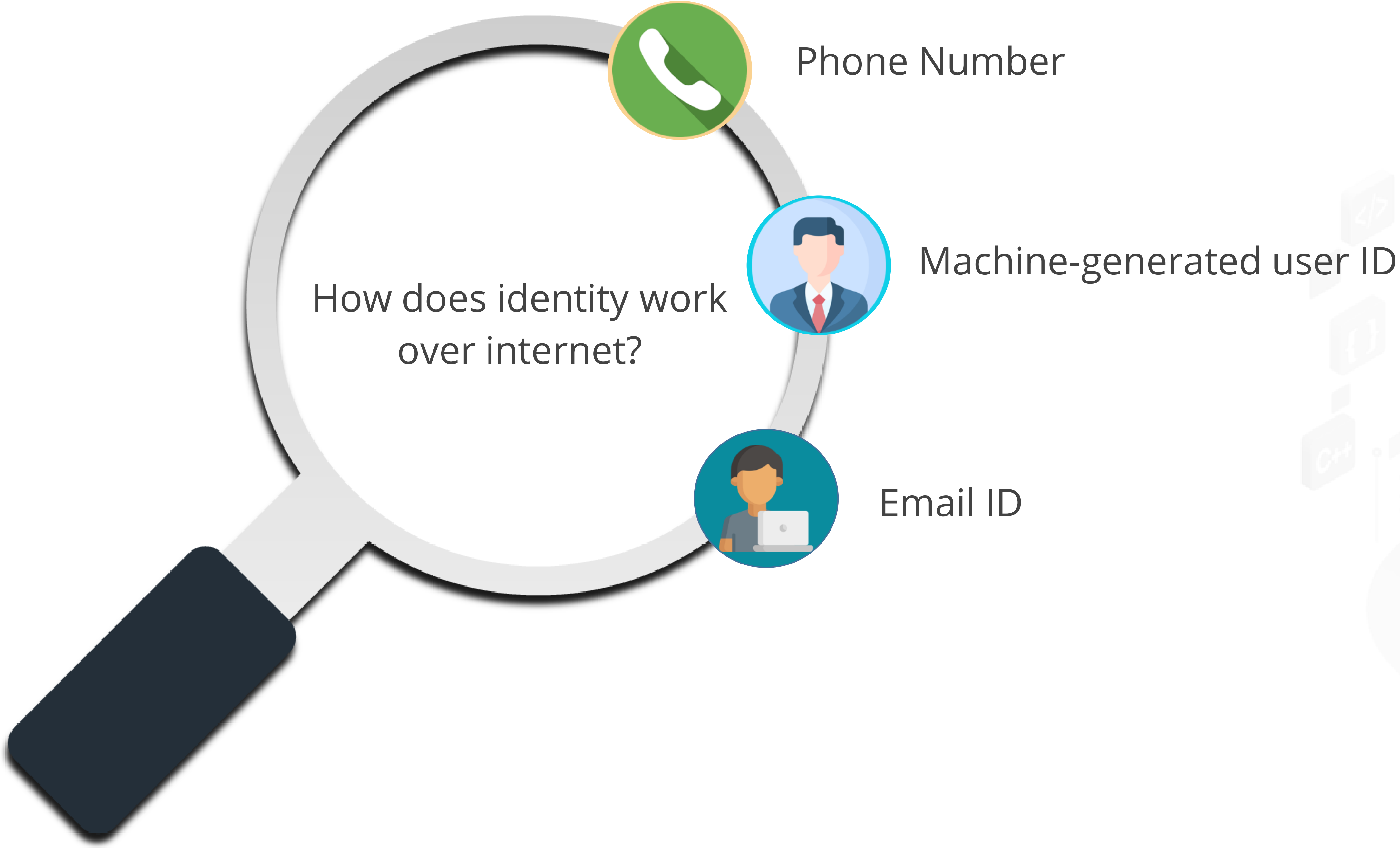
Hyperledger Indy

Importance of Identification in Real World

How does identity work in the real world?

- Recognition of family, employers, and co-workers
- Identification of unknown people
- Establishment of a person's level of access liabilities
- Higher bodies, such as governments and corporations, issue them

Identification on the Internet



How does identity work over internet?

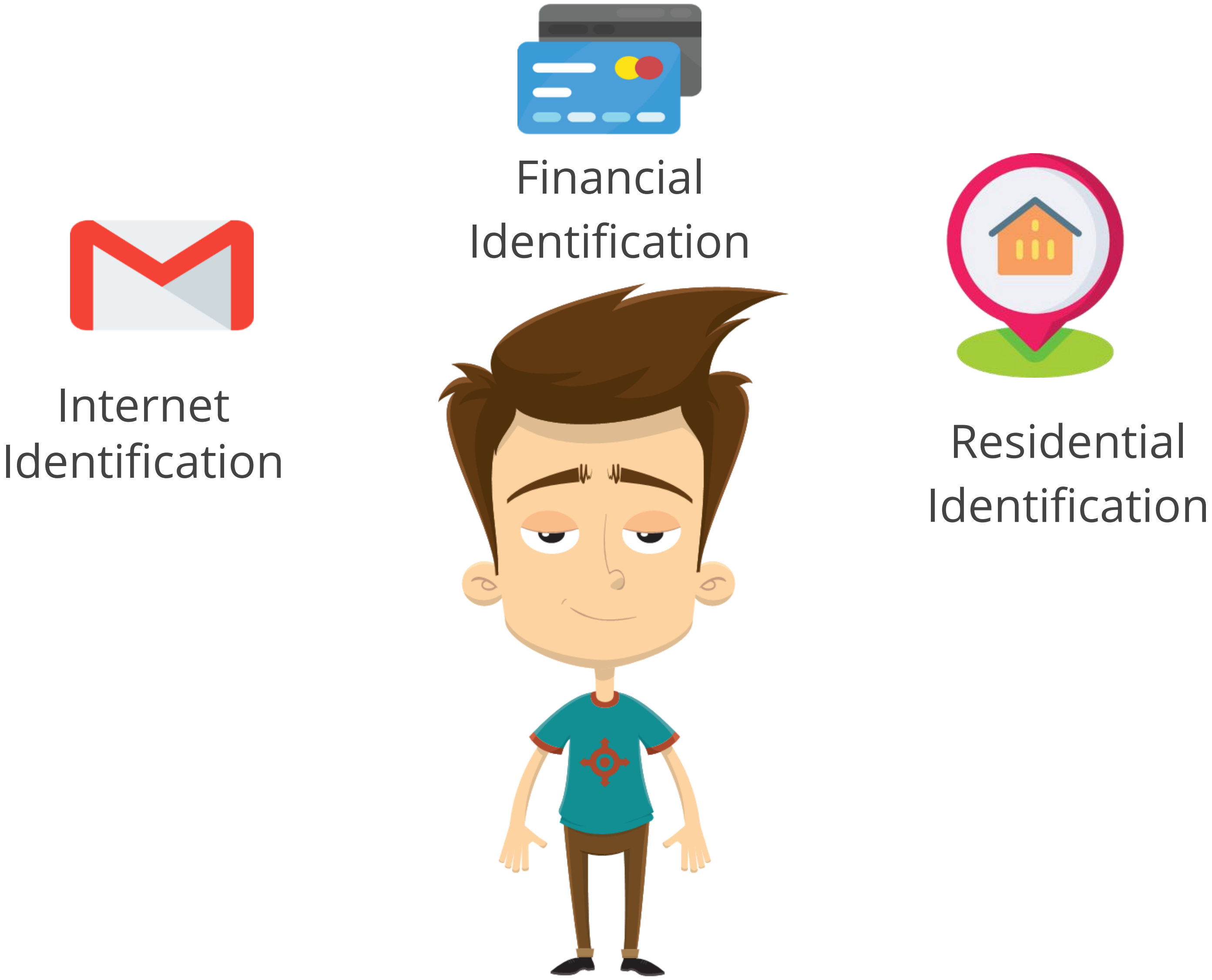
Phone Number

Machine-generated user ID

Email ID

Identification Attributes

Identification allows the user to monitor the type and amount of personal data posted, which is an essential feature for users' informational autonomy.



Financial Identification

Internet Identification

Residential Identification

Identity Based on Risk

Identity attributes are used in two main categories for transactions.

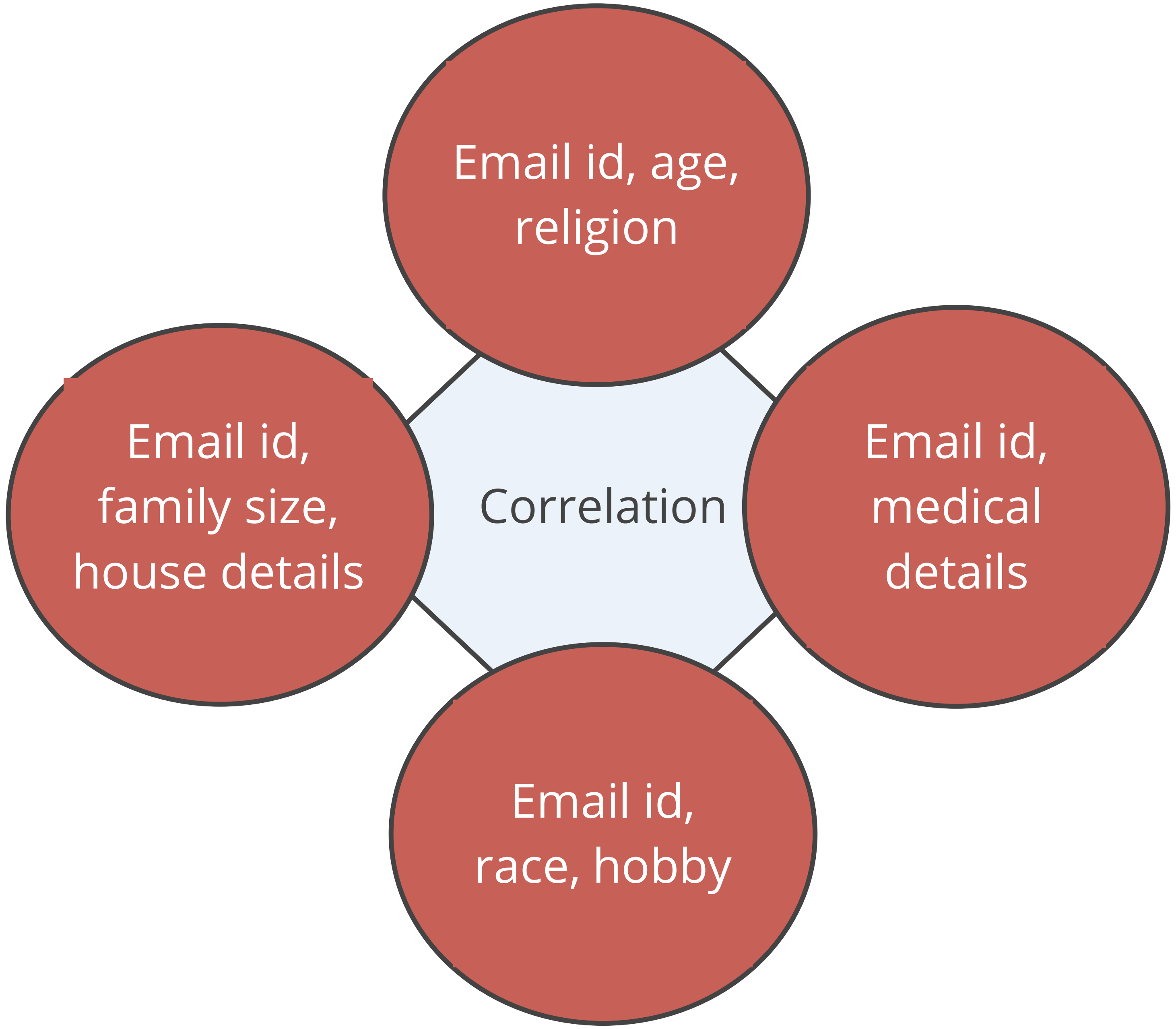
Low Risk

When you buy something online

High Risk

When you open a bank account online

Identity Correlations



Email id, age, religion

Email id, family size, house details

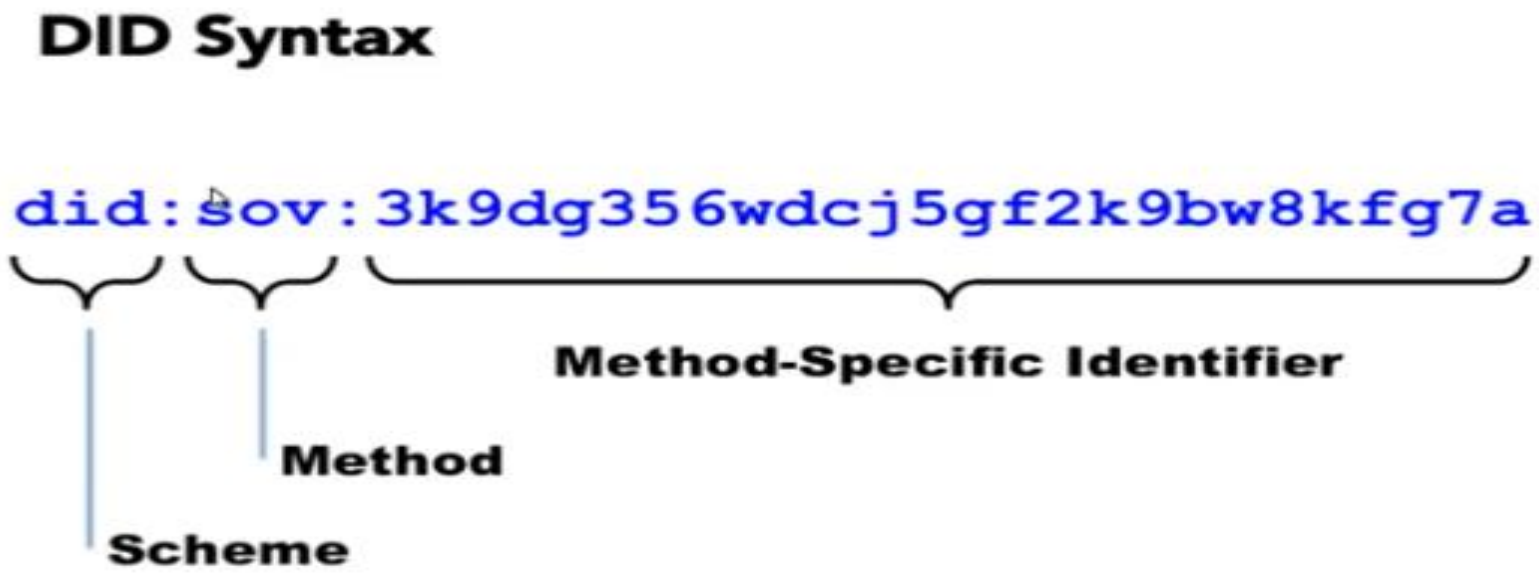
Correlation

Email id, medical details

Email id, race, hobby

Decentralized Identifiers

Decentralized Identifiers (DIDs) are the global identifiers generated by the owner and not by any central authority. They are generated using public and private keys where owner keeps the private key secret.



DID Syntax

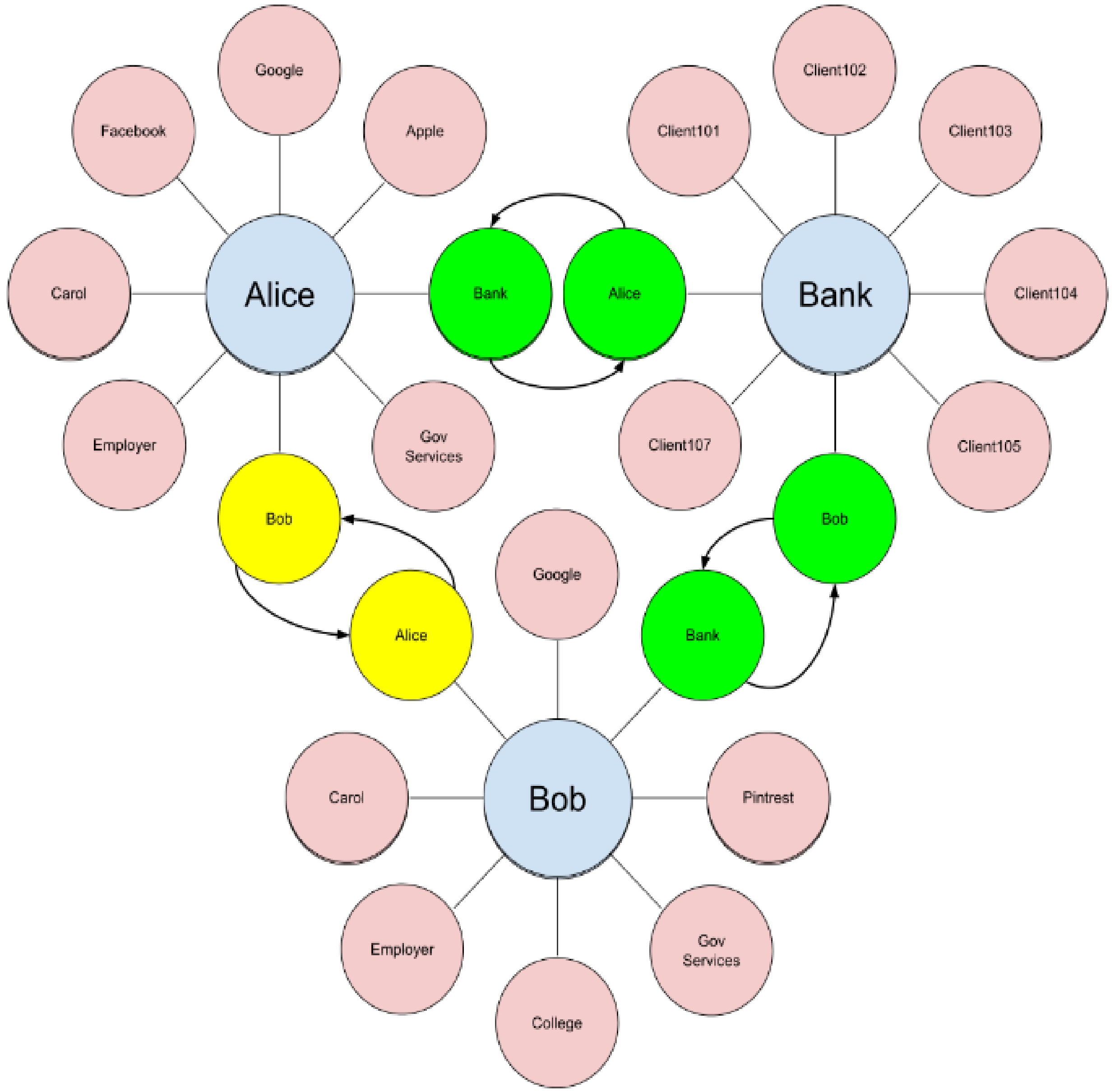
did:sov:3k9dg356wdcj5gf2k9bw8kfg7a

Method-Specific Identifier

Method

Scheme

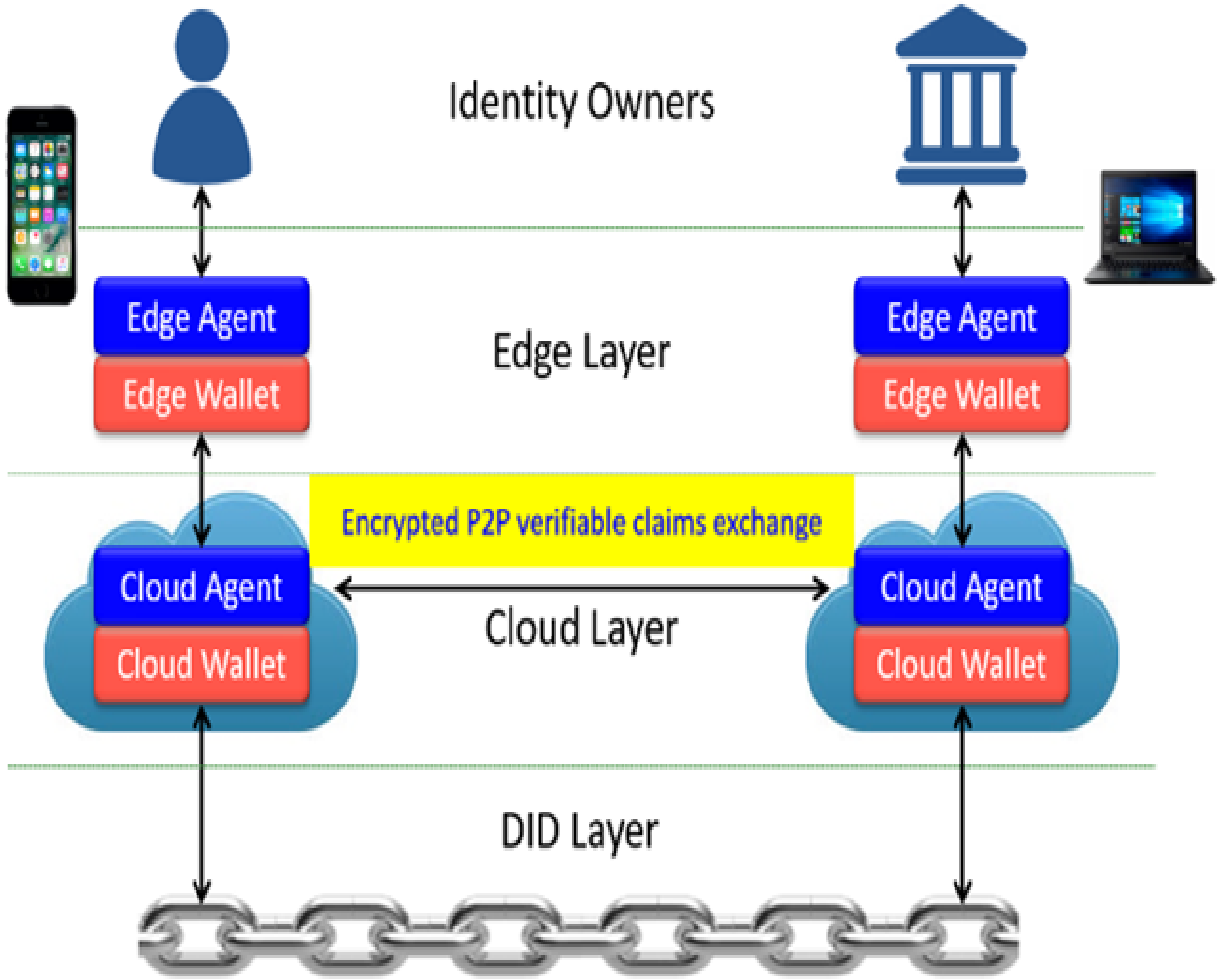
Managing DIDs



At a given time, a user can have many DIDs

- For instance, Alice uses different DIDs for Facebook, Google, Amazon, and banks.
- Similarly banks have many customers so they need to manage their DIDs.

Wallets and Agents



To manage DIDs, Indy makes use of agents and wallets

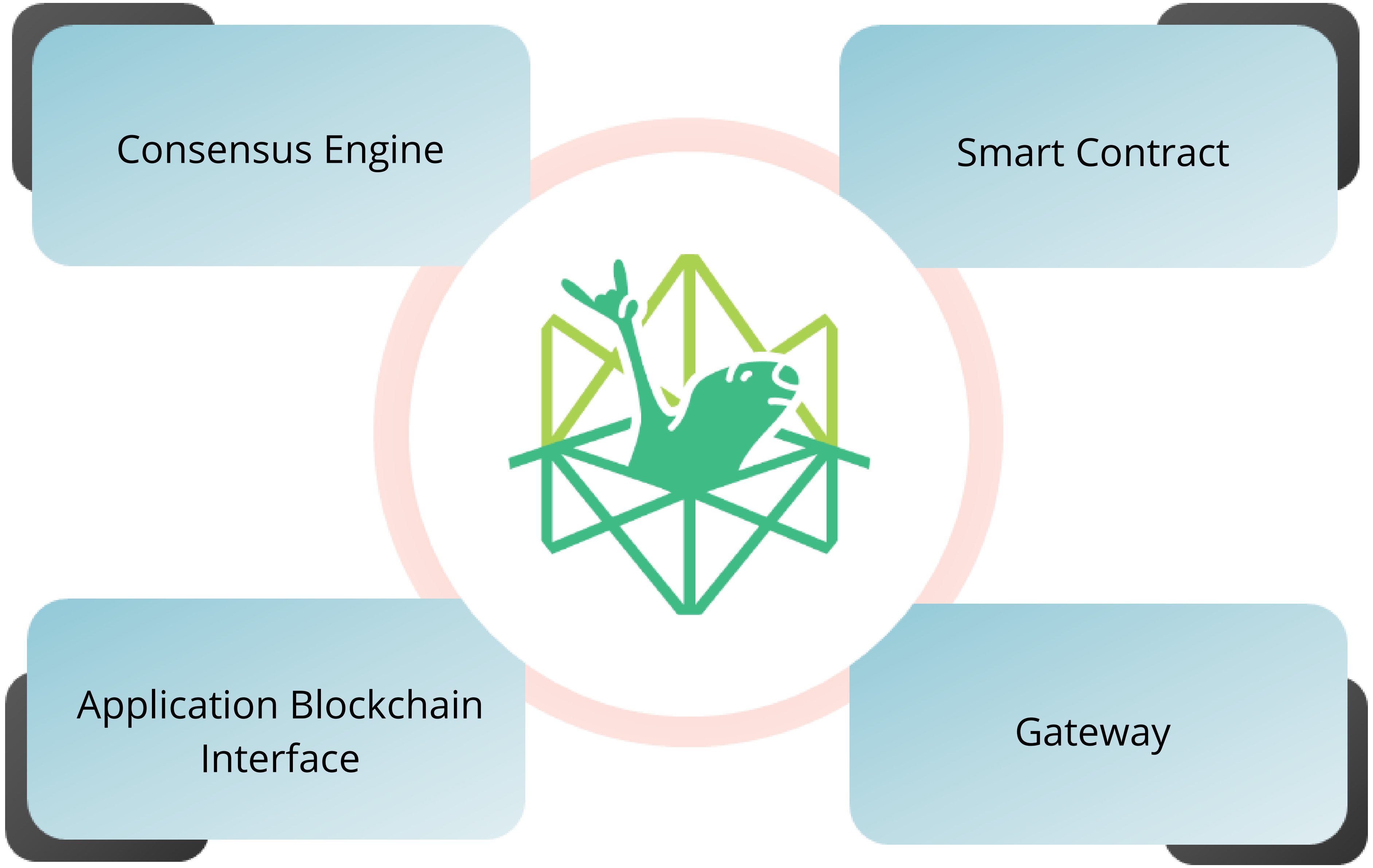
- **Agent:** Software that helps you to interact with other DIDs.
- **Wallet:** It stores DIDs and related information like public keys.

Hyperledger Burrow

Hyperledger Burrow

Hyperledger Burrow is a permissioned Blockchain node built for a multi-chain universe that executes smart contracts following the Ethereum specifications. It uses Ethereum Virtual Machine.

Burrows Features



Consensus Engine

Smart Contract

Application Blockchain Interface

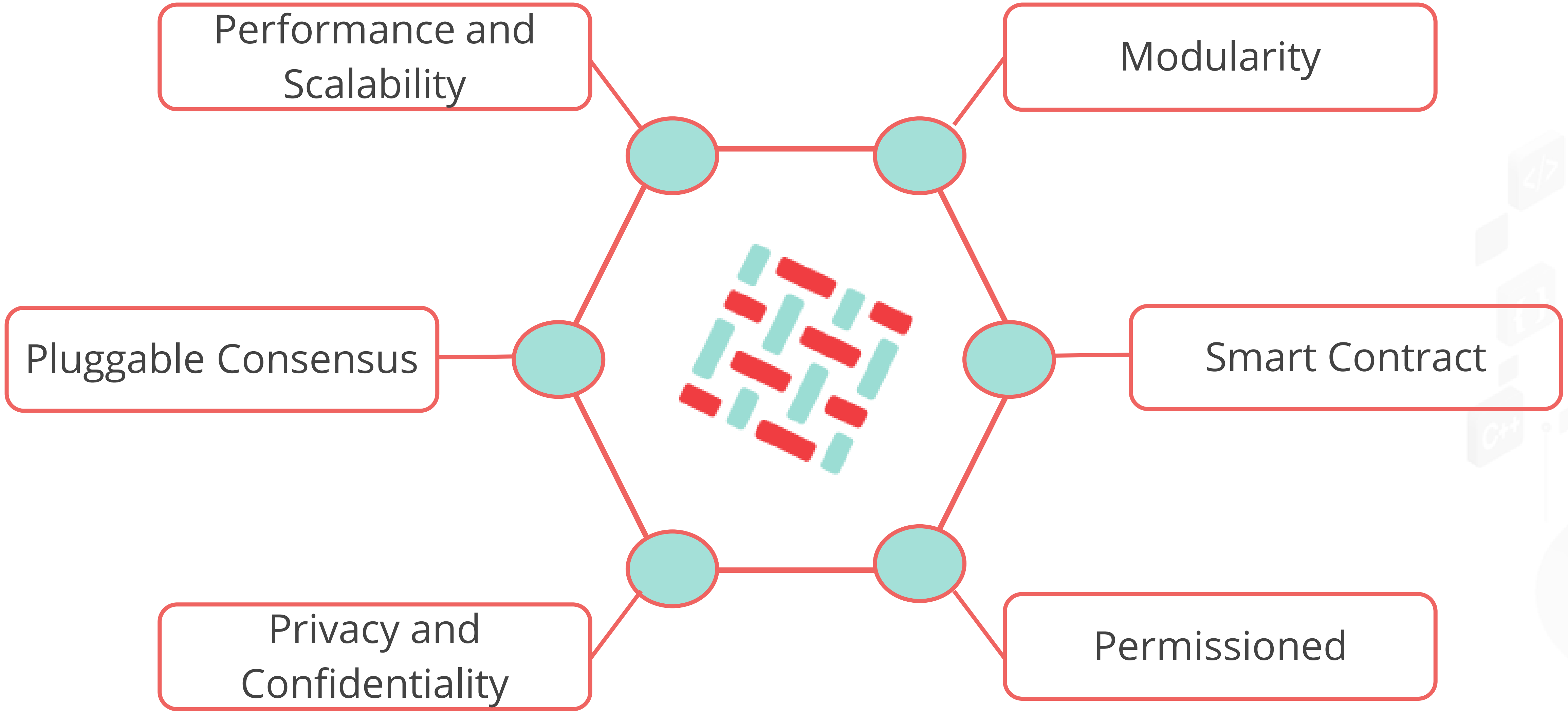
Gateway

Hyperledger Fabric

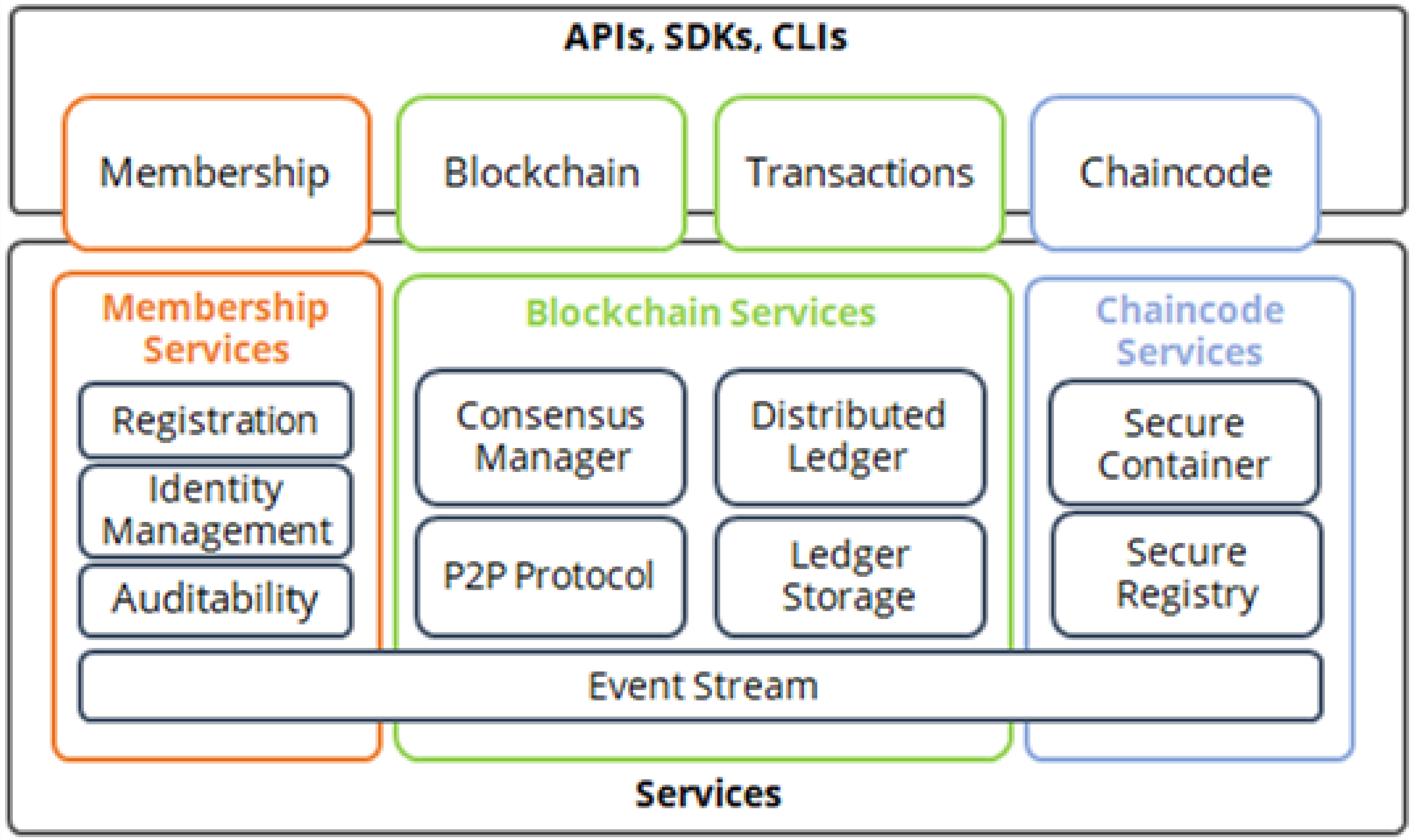
Hyperledger Fabric

Hyperledger Fabric is the first distributed ledger platform to support smart contracts written in general-purpose programming languages rather than domain-specific languages (DSL).

Hyperledger Fabric Features

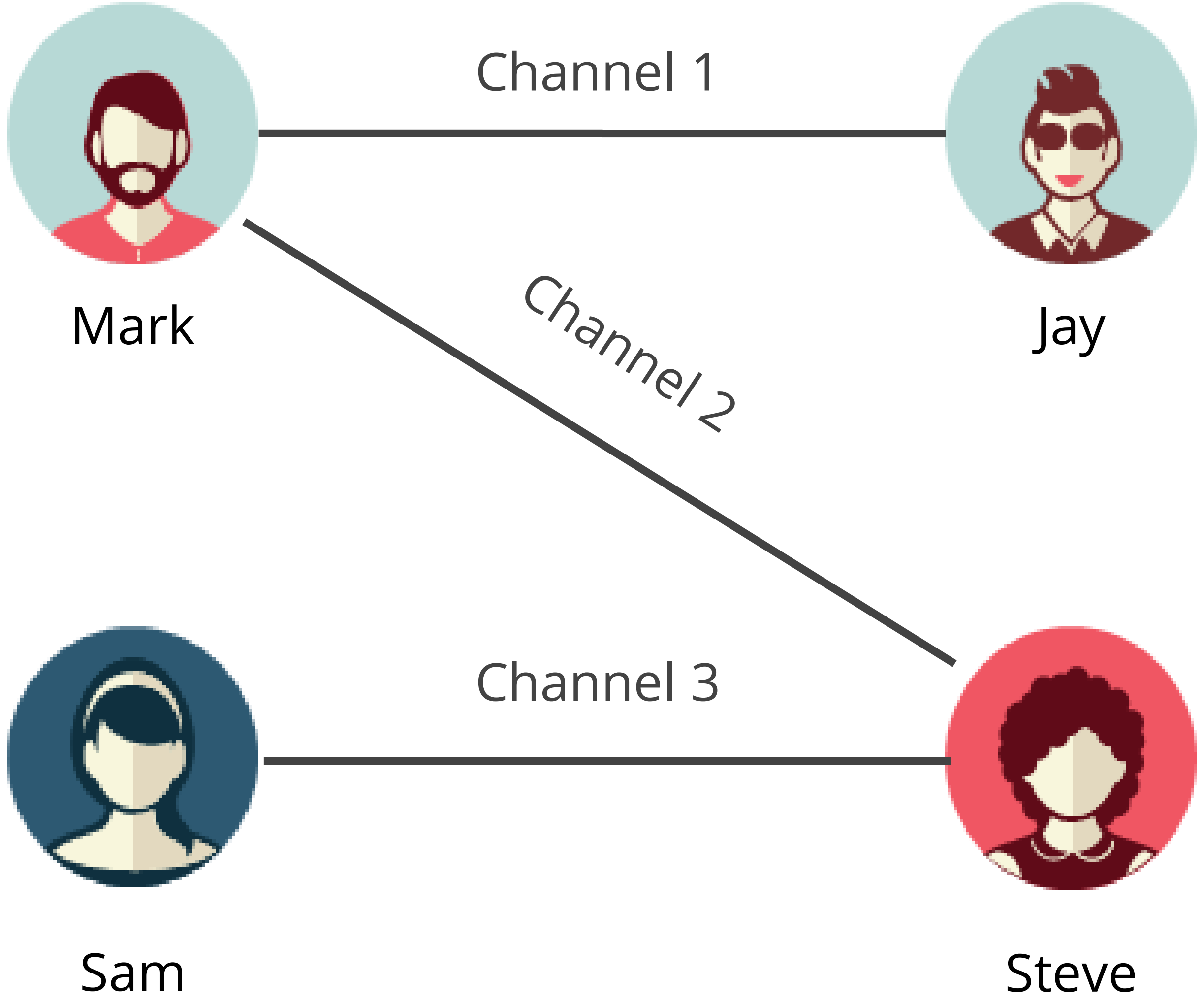


Hyperledger Fabric Architecture



Channel

Channel is private communication between two or more participants in a network.



Membership Service Provider

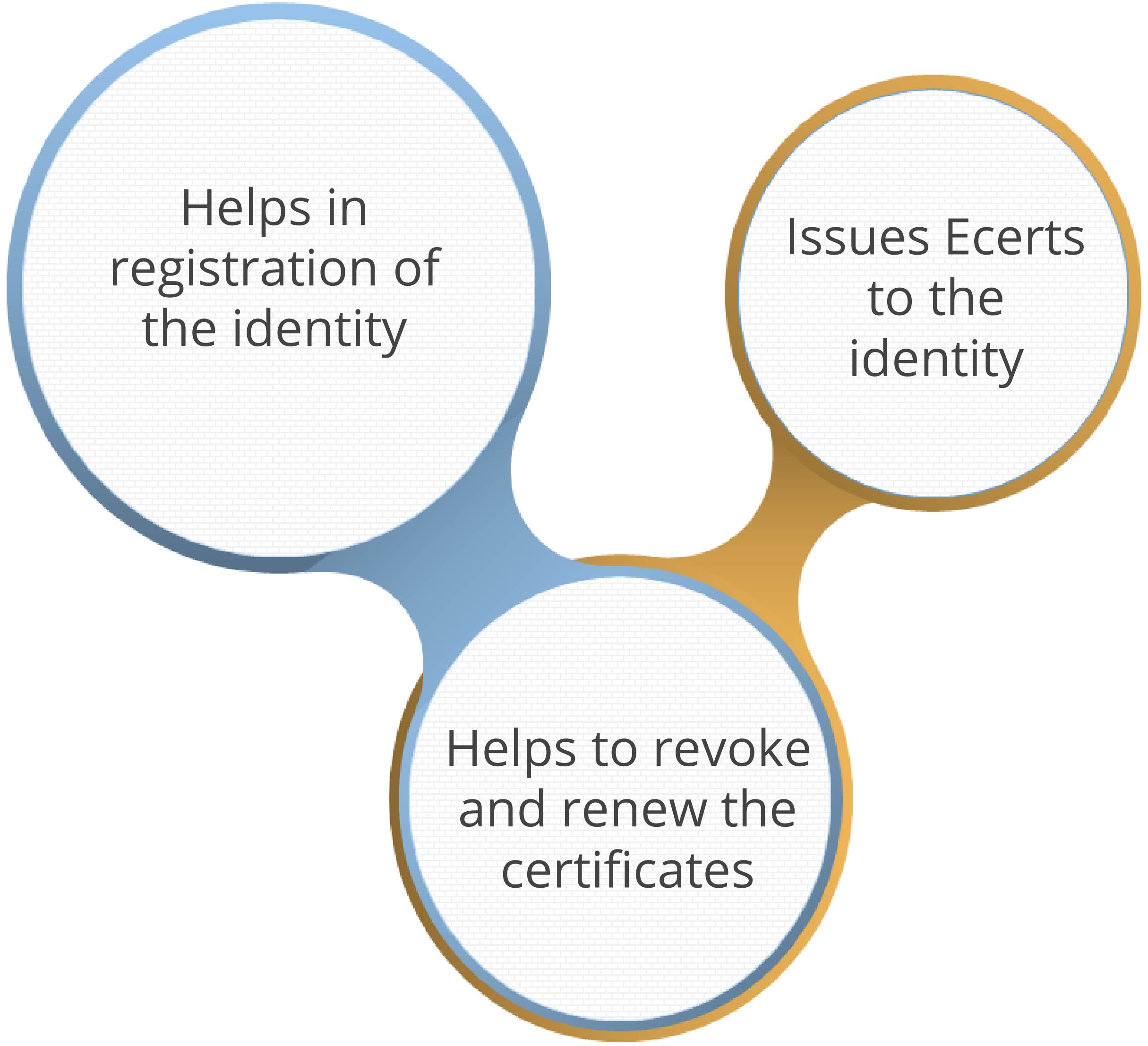
One of the features of the Enterprise Blockchain is that user should not be anonymous rather identity of the user should be known.

Contains a list of parties/actors involved in Blockchain network

Provides identity to each participant

Fabric CA

Fabric CA is a certificate authority component which can integrate existing registries like LDAP. MSP leverages fabric CA component.

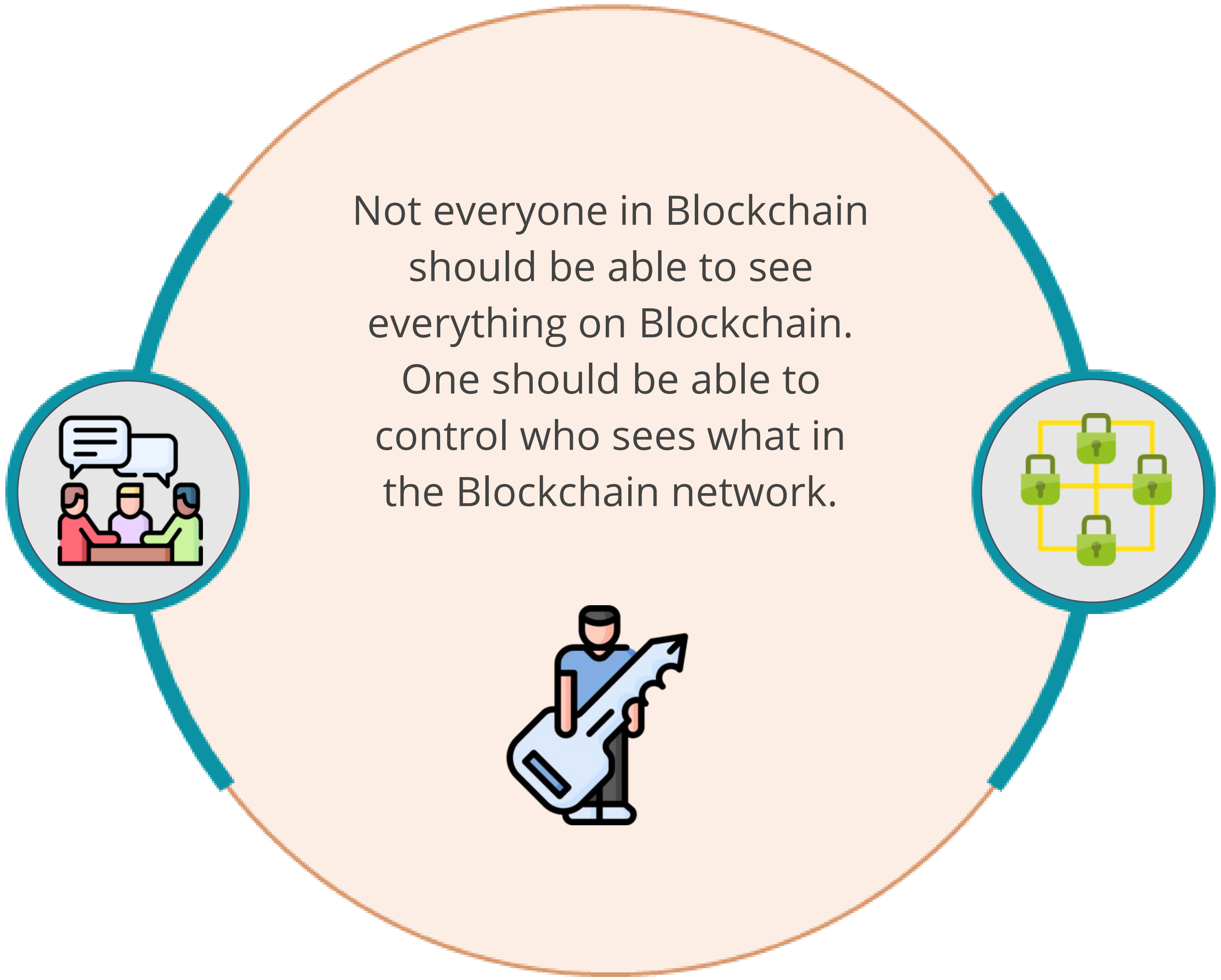


Helps in registration of the identity

Issues Ecerts to the identity

Helps to revoke and renew the certificates

Access Control List



Not everyone in Blockchain should be able to see everything on Blockchain. One should be able to control who sees what in the Blockchain network.

Handle permission for parties involved

Handle permission at channel level as well

Types of Nodes

Committing Nodes

In a Hyperledger Fabric only a few nodes keep a copy of ledger, these nodes are known as Committing Nodes.

Endorsing Nodes

In a Hyperledger Fabric only a few nodes execute chaincode, these nodes are known as Endorsing Nodes.

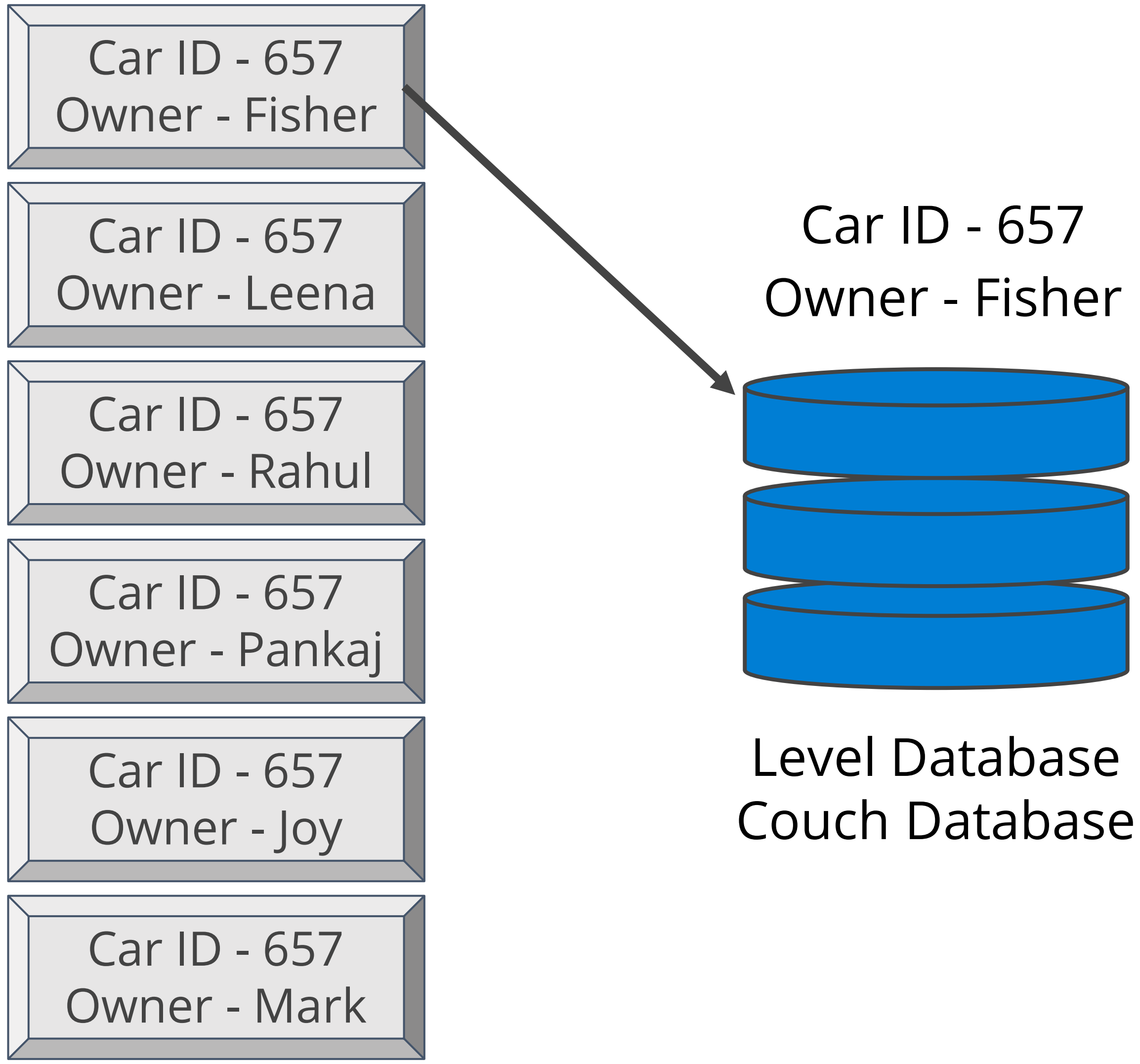
Ordering Nodes

The Ordering Node is responsible for maintaining the transaction sequence in Blockchain.

Advantage of Hyperledger Fabric

- A wide range of industries will benefit from the Hyperledger fabric project.
- Since the Hyperledger framework is open source, anybody can use it to boost their business.
- Since the Hyperledger fabric project is absolutely modular, you can use as many Hyperledger functionalities as you like.

State Database



- Hyperledger Fabric is faster compared to Public Blockchain due to State Database.
- State Database is of two types, Level Database and Couch Database.

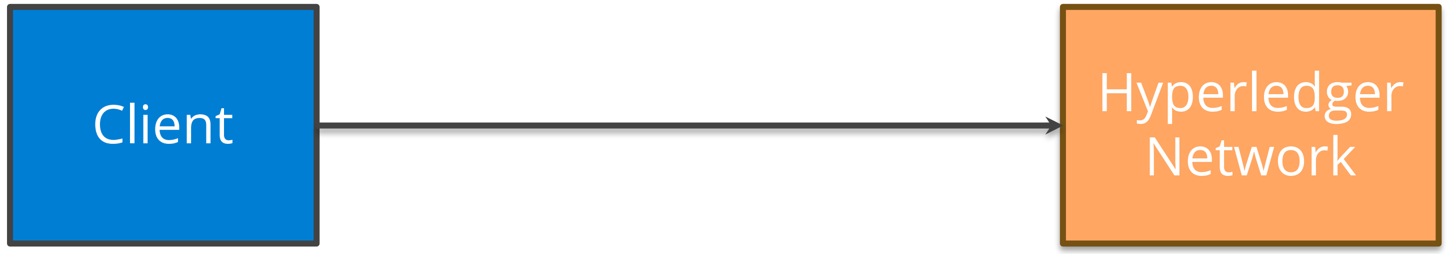
TECHNOLOGY

Hyperledger Fabric Transaction

Hyperledger Fabric Transaction

Step 1: Client initiating the transaction

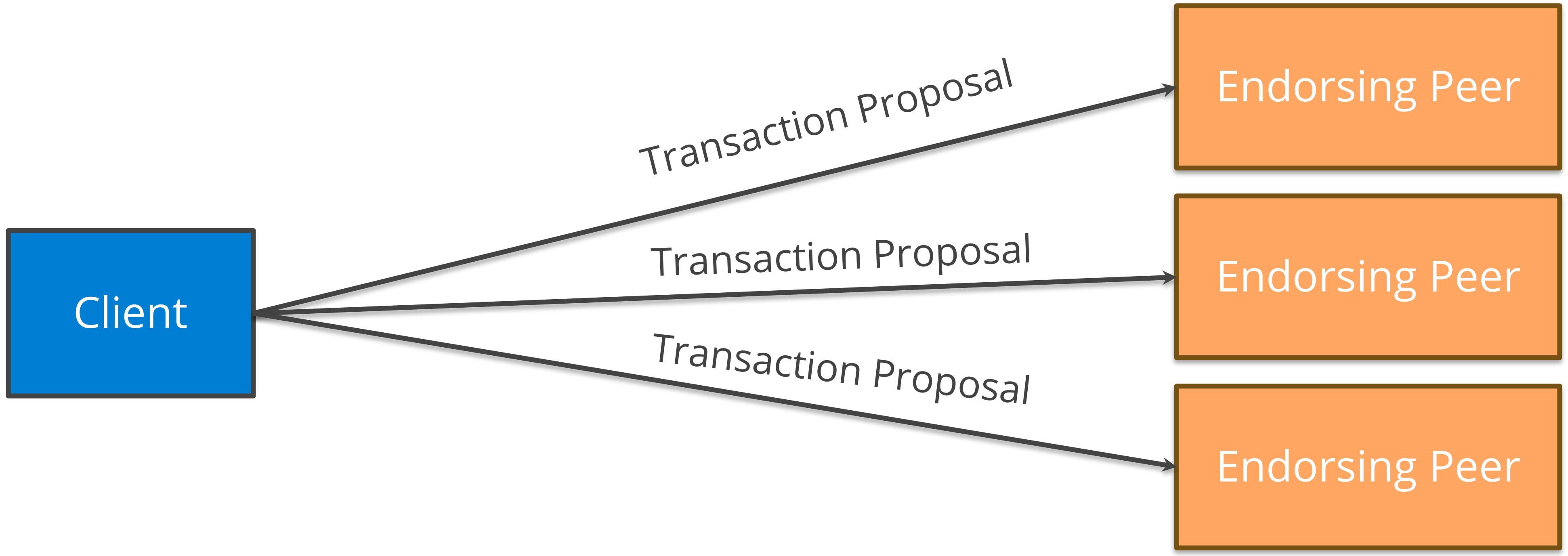
- Network verifies the client's identity using Membership Service Provider
- Network uses an Access Control List to determine if the requester has permission to access the network



Hyperledger Fabric Transaction

Step 2: Validate the transaction

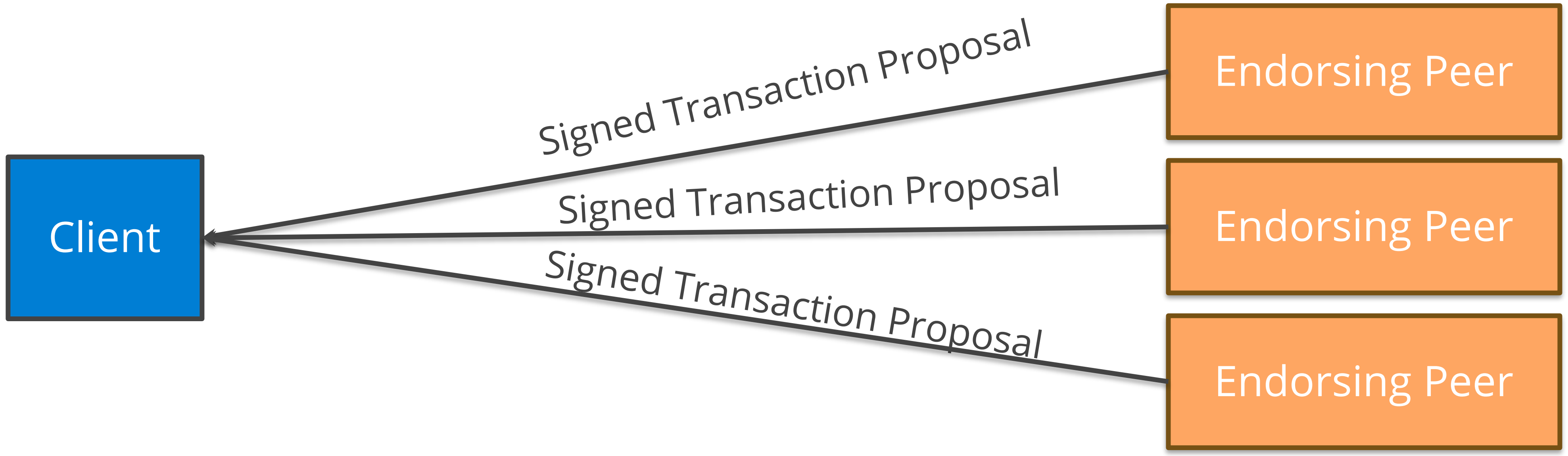
- Endorsing Peers are notified of the transaction through a proposal



Hyperledger Fabric Transaction

Step 3: Simulating the transaction

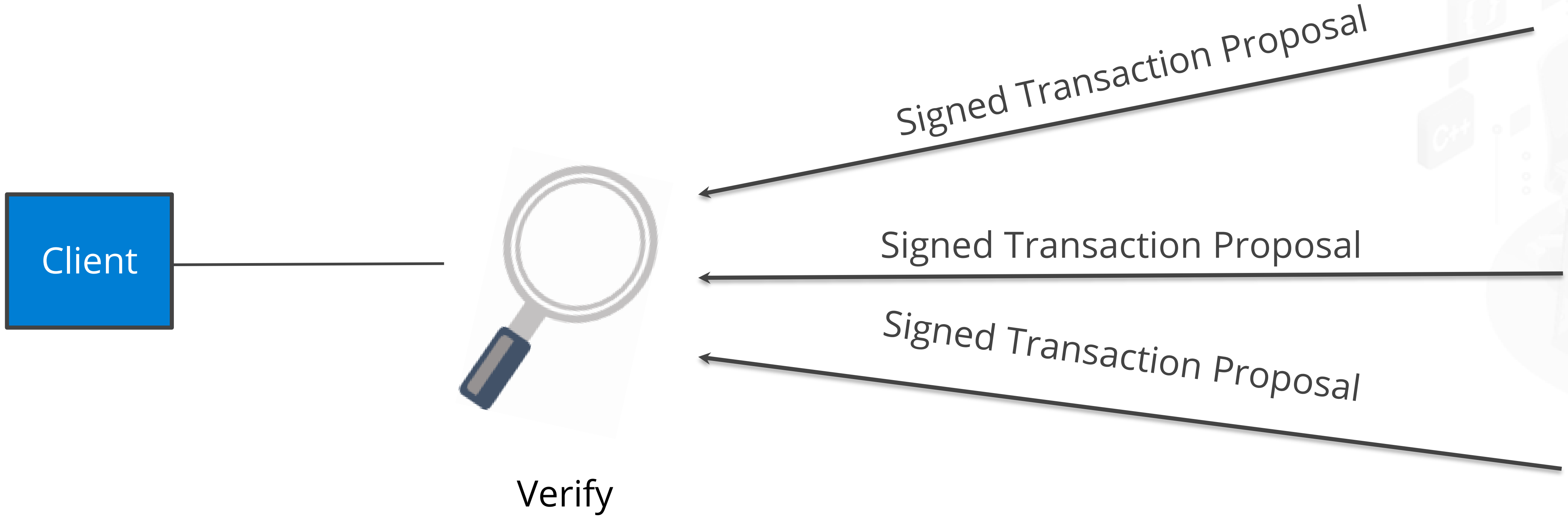
- Endorsing Peers execute the chaincode and report back to the client program


Client
Signed Transaction Proposal
Signed Transaction Proposal
Signed Transaction Proposal
Endorsing Peer
Endorsing Peer
Endorsing Peer

Hyperledger Fabric Transaction

Step 4: Verifying proposal response

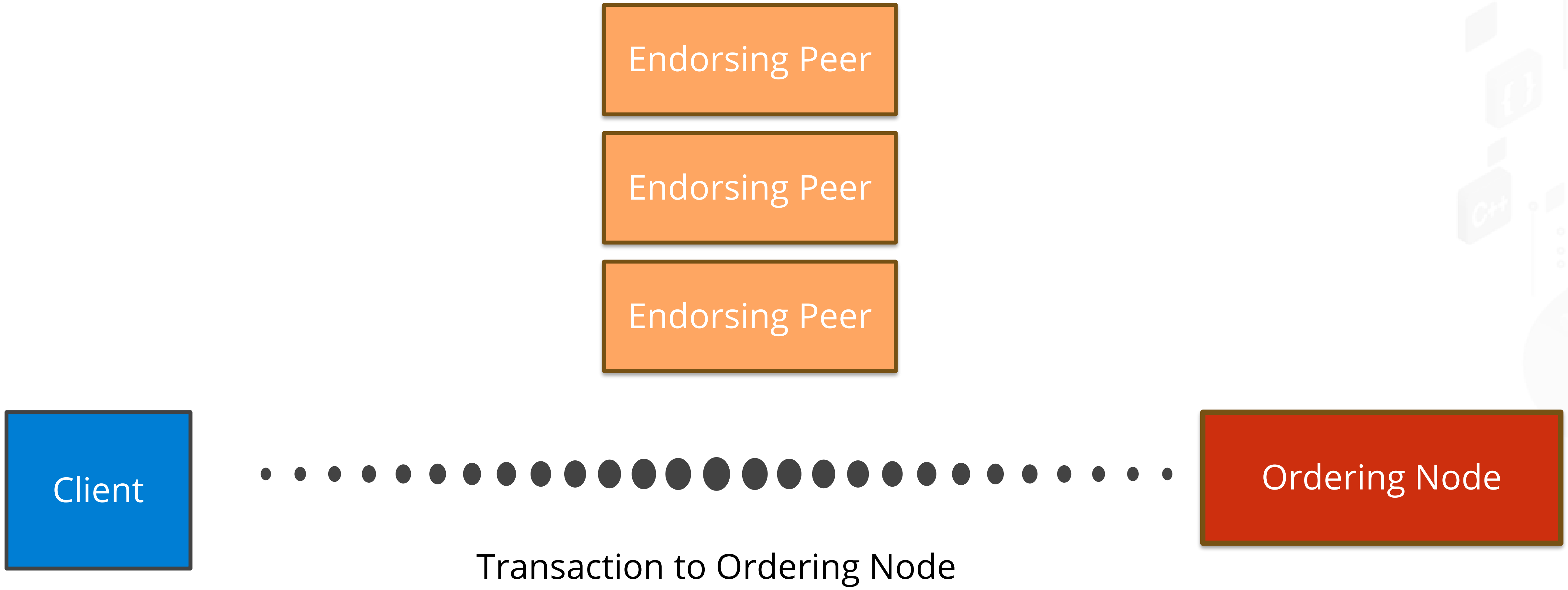
- The client application receives the response from Endorsing Peers and checks the response to see if consensus has been achieved


Signed Transaction Proposal
Signed Transaction Proposal
Signed Transaction Proposal
Client
Verify

Hyperledger Fabric Transaction

Step 5: Broadcast transaction to the order

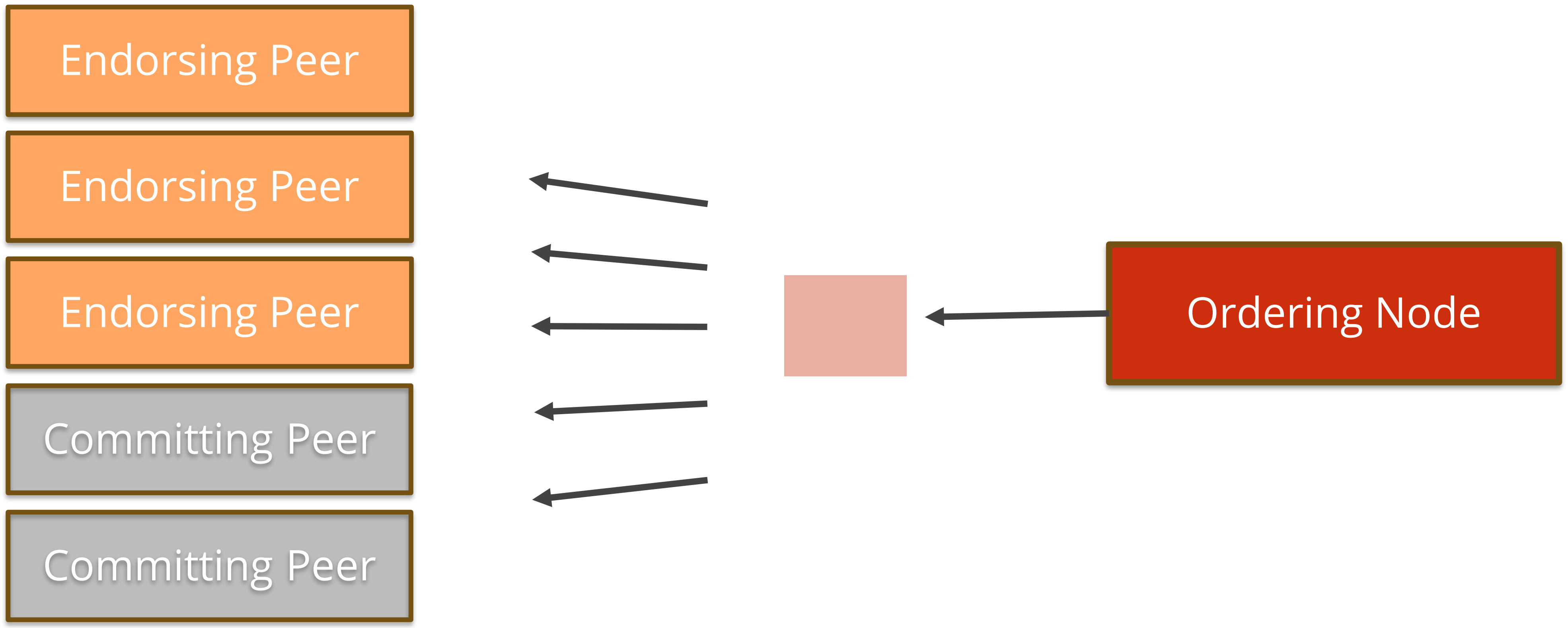
- The Ordering Node is notified by the client application that a new transaction is being recorded on the ledger



Hyperledger Fabric Transaction

Step 6: Order transactions and create block

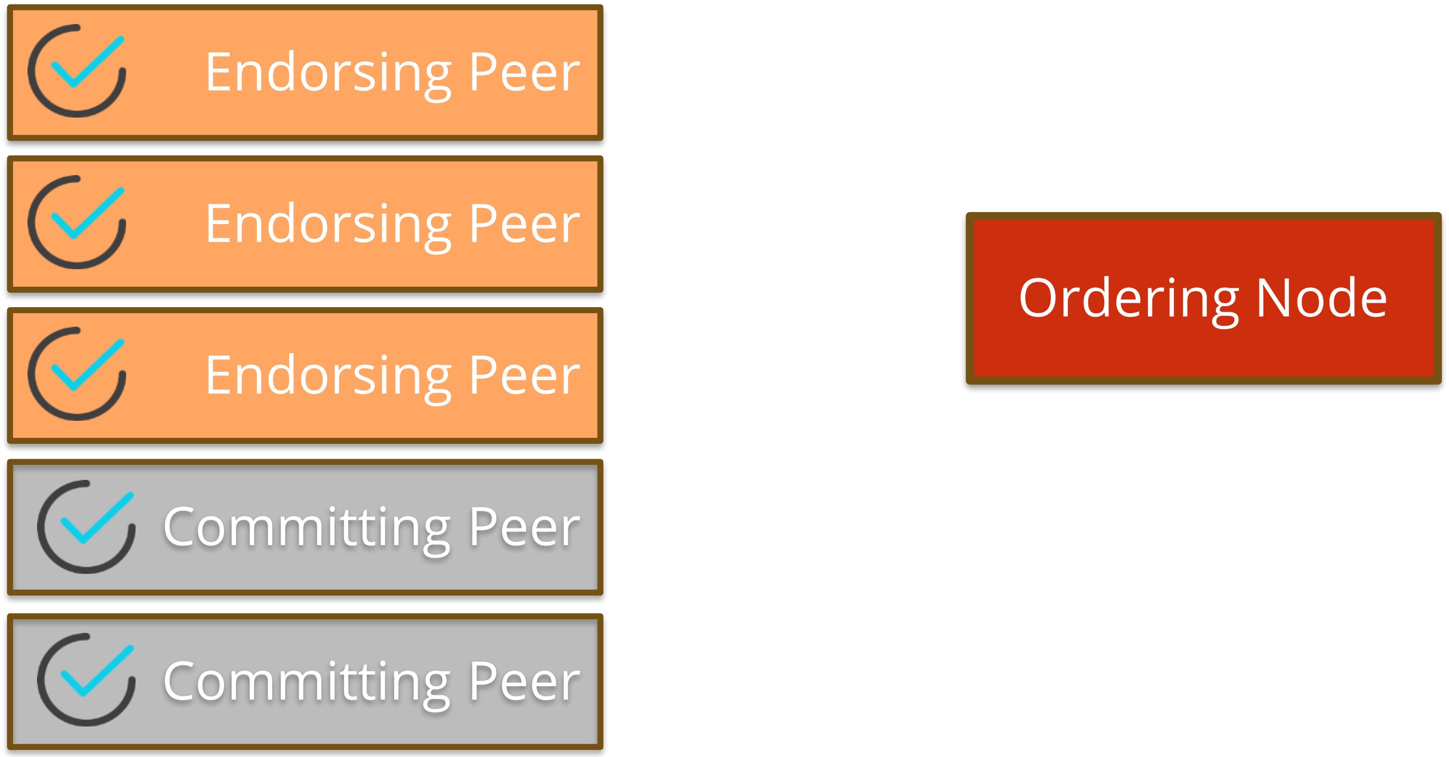
- Ordering Node orders the transactions, generates a block with those transactions, and gives it back to the Endorsing Peers



Hyperledger Fabric Transaction

Step 7: Peers validate each transaction in block

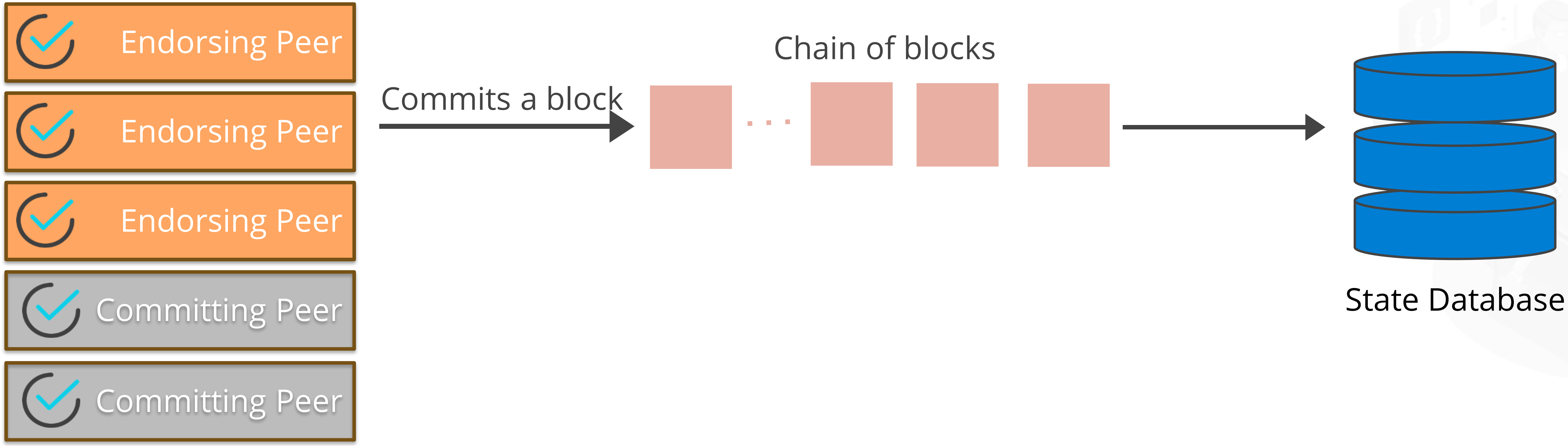
- Peers verify each transaction in the block and notify the client as well



Hyperledger Fabric Transaction

Step 8: Committing to Ledger

- Committing nodes (and the Endorsing nodes) record the copy of the record in their ledger



Powered by simplilearn

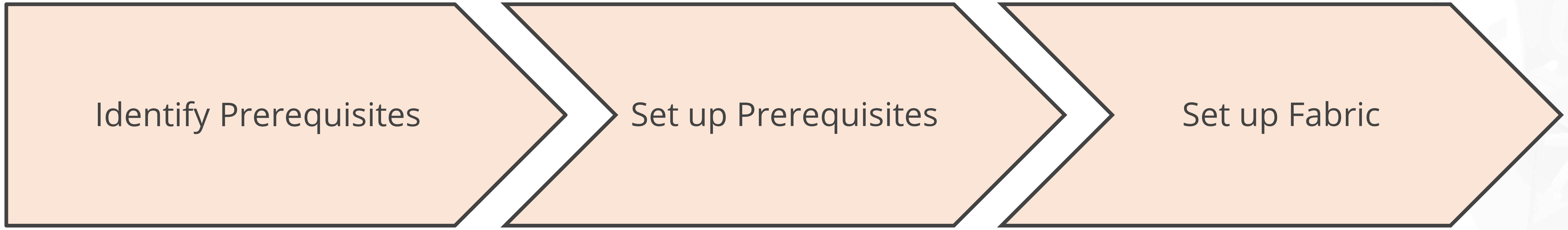
IIT KANPUR
Indian Institute of Technology, Kanpur

TECHNOLOGY

Fabric Network

Hyperledger Fabric setup

Hyperledger setup involves the following high-level steps:



Identify Prerequisites

Set up Prerequisites

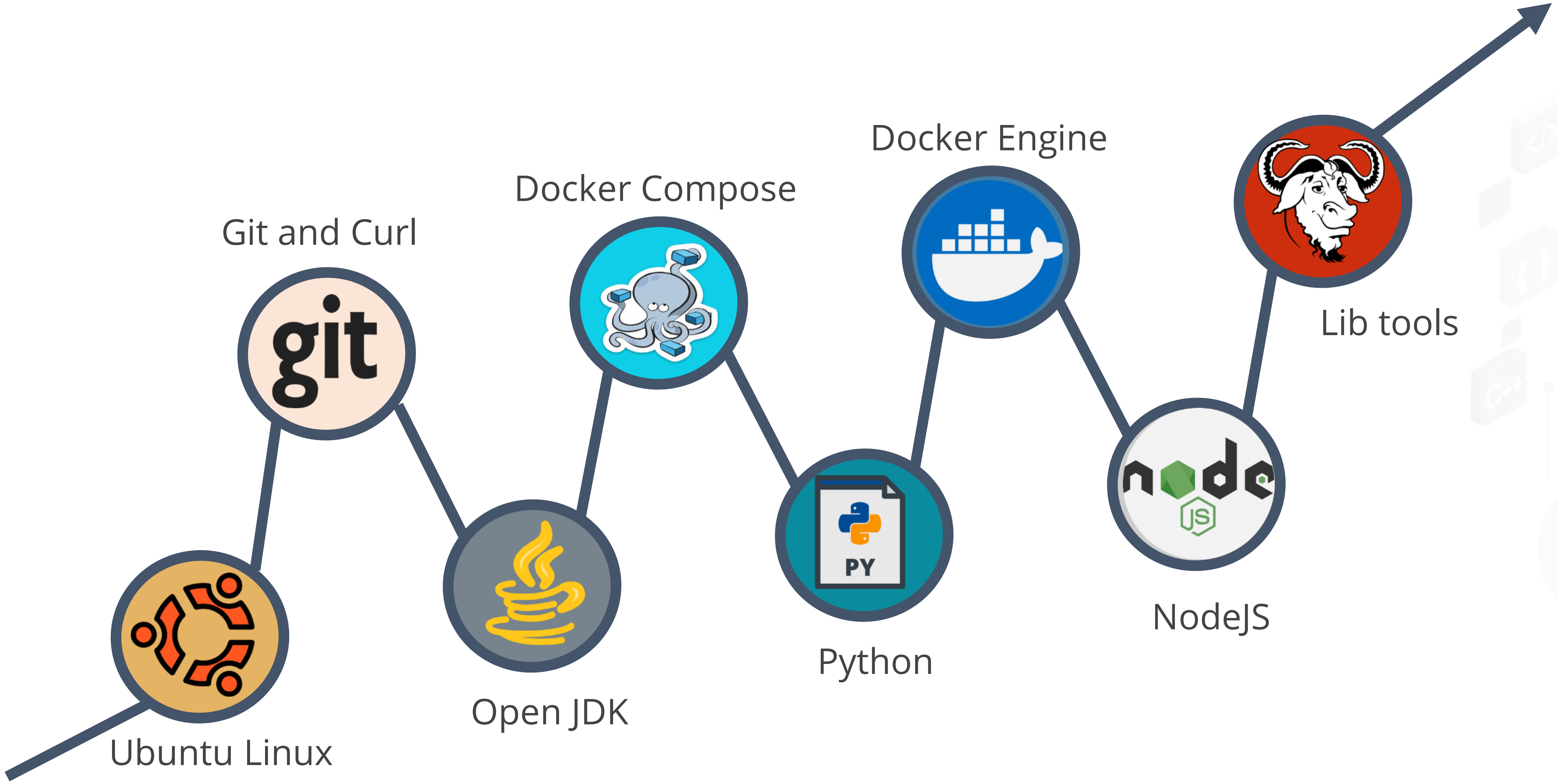
Set up Fabric

Powered by simplilearn

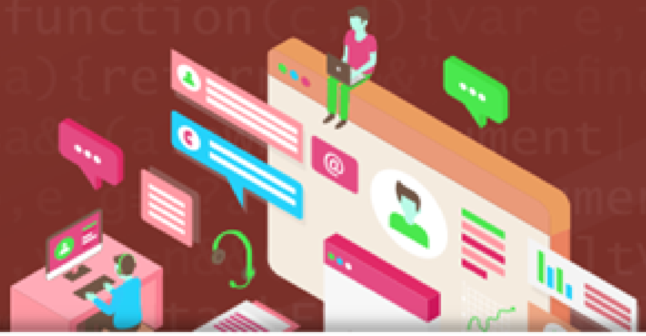
IIT KANPUR
Indian Institute of Technology, Kanpur

Identify Prerequisites

The following Operating System, Software, and tools are required to run Hyperledger in your System:



Set up Hyperledger Fabric Prerequisites



Problem Statement: You are given a task to install Hyperledger Fabric prerequisites.

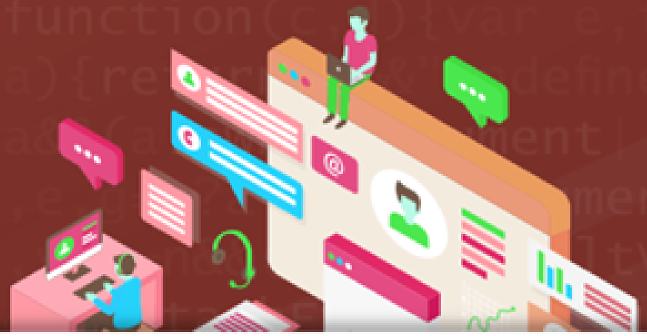
ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to set up Hyperledger Fabric prerequisites:

1. Installing Curl in the local machine
2. Installing Node.js in the local machine
3. Installing Git in the local machine
4. Installing Python in the local machine
5. Installing Lib tools in the local machine
6. Downloading and installing Docker CE in the local machine
7. Setting up Docker Compose in the local machine

Set up Hyperledger Fabric



Problem Statement: You are given a task to install Hyperledger Fabric.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to set up Hyperledger Fabric:

1. Cloning the Hyperledger Fabric repository and installing it

TECHNOLOGY

Fabric Network Types

Identify Prerequisites

Development network

The development network is mainly used for chaincode unit testing and for ensuring the functionality of chaincode operates as intended.

Test network

The test network is a production network replica which has the same rules as those of the production.

Enterprise network

The enterprise network is the same as the production network. It has nodes and links that are spatially interconnected.

Fabric Network Files

crypto-config.yaml

configtx.yaml

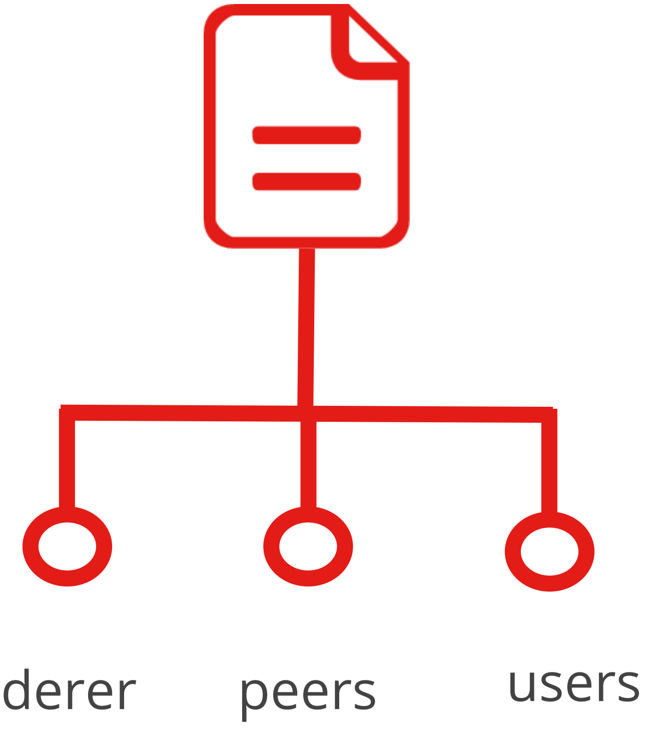
docker-compose.yaml

- Fabric Network files account for the majority of the development, test, and production networks.
- These files can be updated as required by the network administrator.

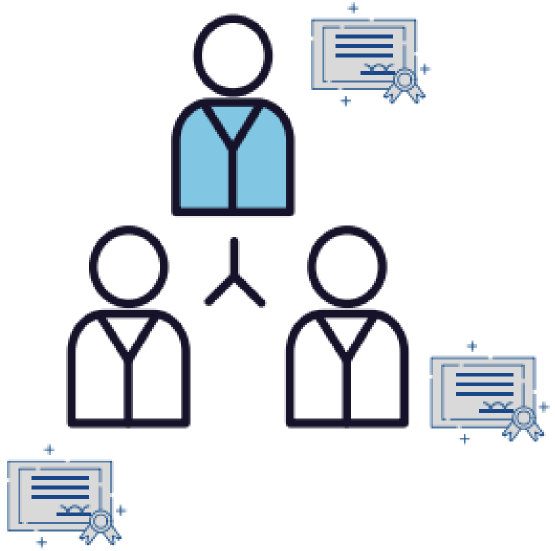
Crypto-config.yaml

This file assists the users to generate public and private keys, digital certificates for peers, and ordered service using the command cryptogen.

Structure



Purpose



Configtx.yaml

This file assists to generate genesis block, which is an appropriate block in Blockchain.

Structure

Organization orders

Application capabilities

Profiles

Purpose

🟩 Genesis block

Channel artifacts

Docker-compose.yaml

Docker containers are used to run the Hyperledger Fabric network. This file contains information about each Docker container, such as name, image, port, and volumes.

Structure

CA containers

Peer and order containers

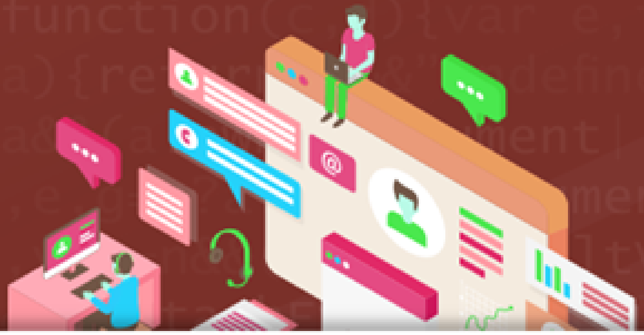
State database containers

Purpose

docker Create containers

Start and stop network

Start and Stop Test Network



Problem Statement: You are given a task to start and stop test network.

ASSISTED PRACTICE

Assisted Practise: Guidelines

Steps to start and stop test network:

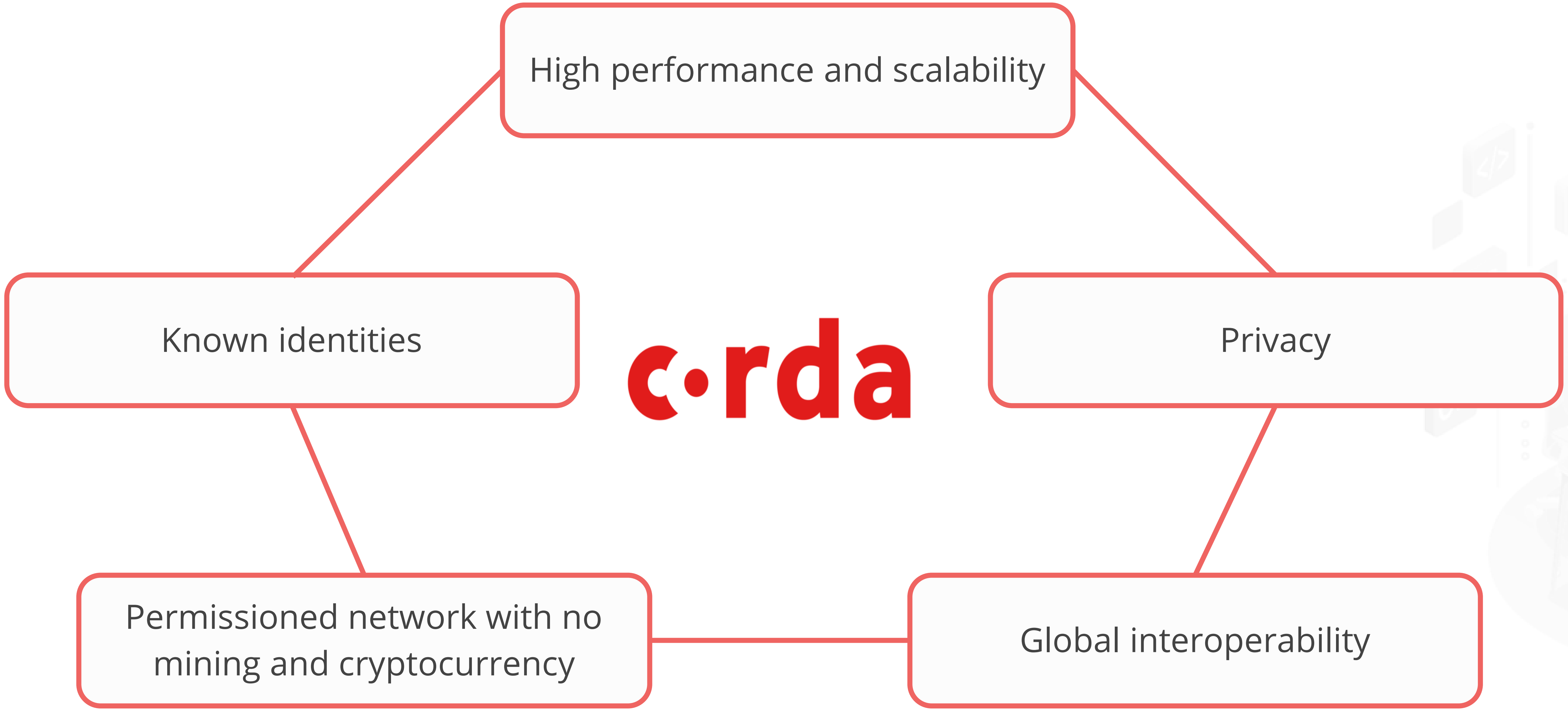
1. Setting up the standard Hyperledger test network
2. Setting up a test network with CA containers
3. Setting up a test network with CouchDB containers
4. Setting up a test network with all the above parameters

R3 Corda

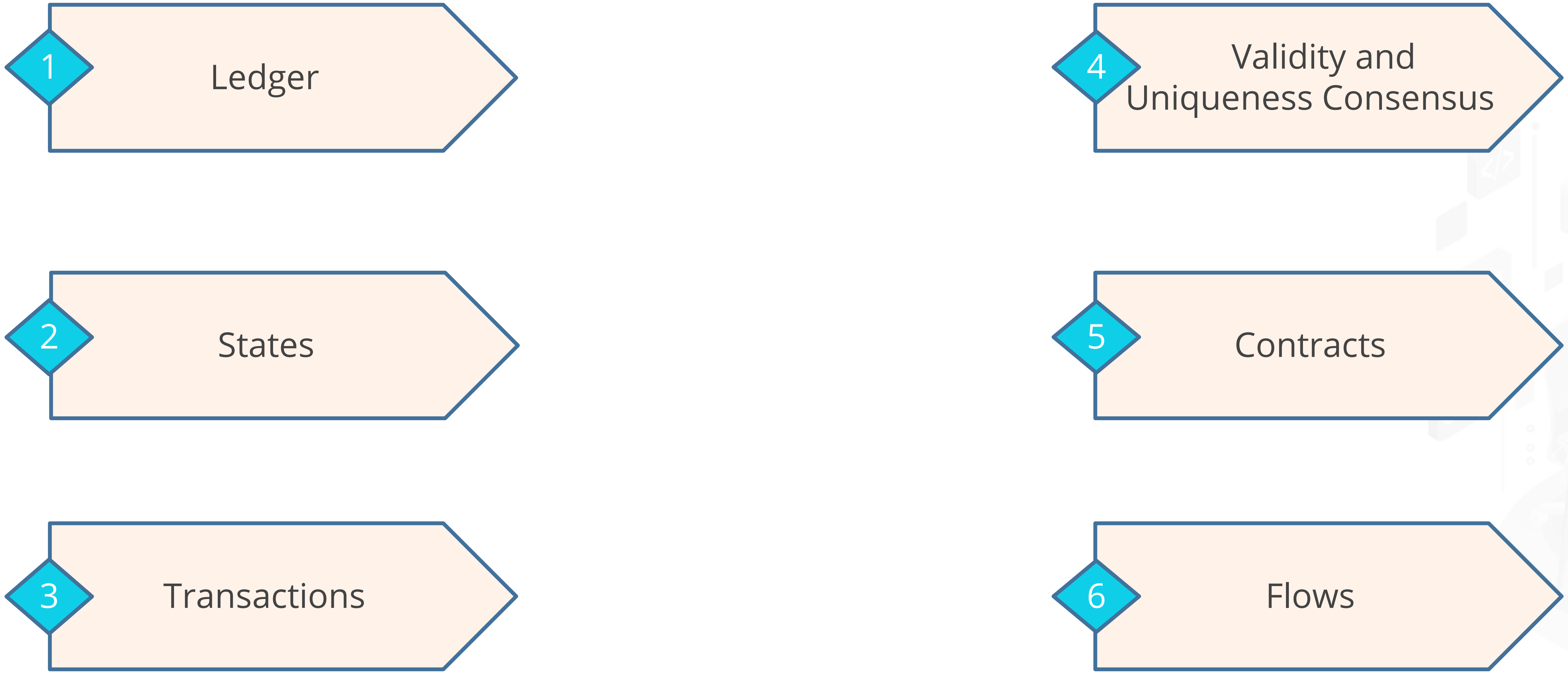
Corda Introduction

Corda is a distributed ledger platform that maintains privacy using smart contracts. It reduces transaction and record-keeping costs and streamlines business operations.

Corda Features


High performance and scalability
Known identities
corda
Privacy
Permissioned network with no mining and cryptocurrency
Global interoperability

Corda Key Concepts



1 Ledger

2 States

3 Transactions

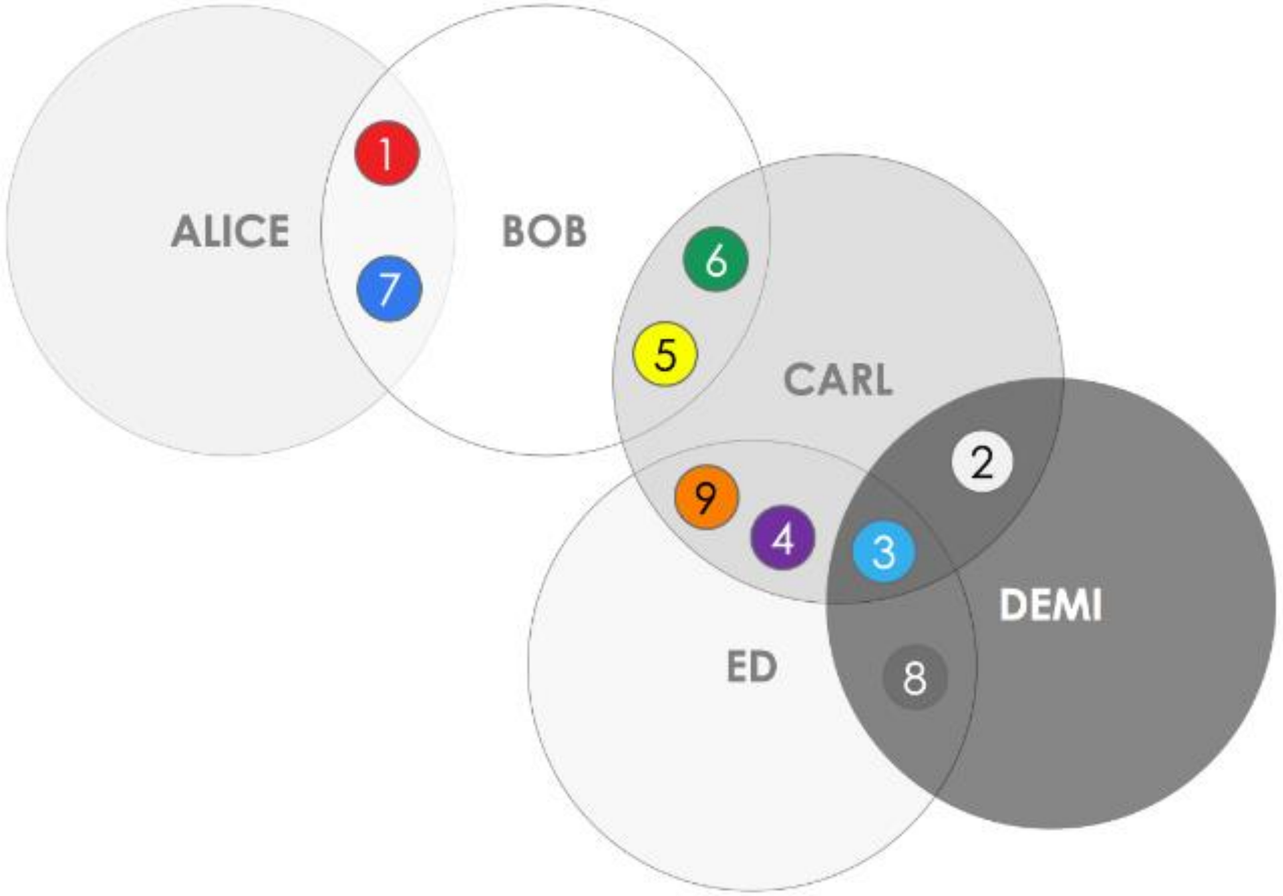
4 Validity and Uniqueness Consensus

5 Contracts

6 Flows

Ledger

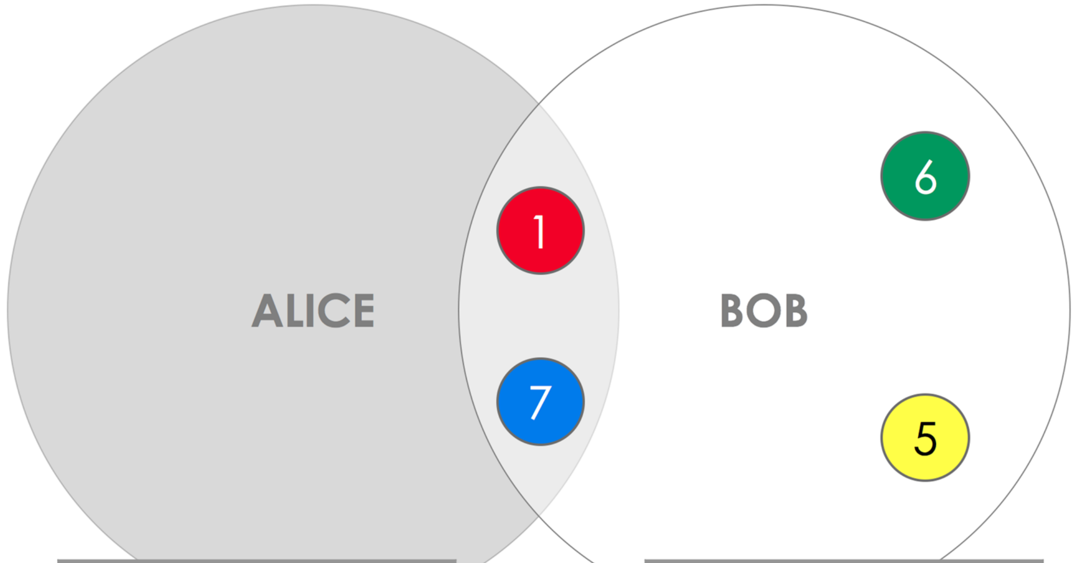
In Corda, every node has its own ledger where that node stores the transactions. This implies there is no central ledger.



NODES	Visibility
ALICE	1,7
BOB	1,5,6,7
CARL	2,3,4,5,6,9
ED	3,4,8,9
DEMI	2,3,8

Reference: *https://docs.corda.net/docs/corda-os/4.6/key-concepts-ledger.html*

Bilateral Ledger



Id	Fact
1	“Much consensus”
7	“So bilateral”
11	“Wow ledger”

Id	Fact
1	“Much consensus”
6	“Very fact”
7	“So bilateral”
5	“amaze network”

- Each node maintains its own ledger.
- Nodes in the Corda network may share data with one another, but ALICE may have data that is not shared with BOB. As an example, set 11
- Data from one node can be shared with other nodes. For example, BOB can share data with ALICE while also sharing it with CARL.

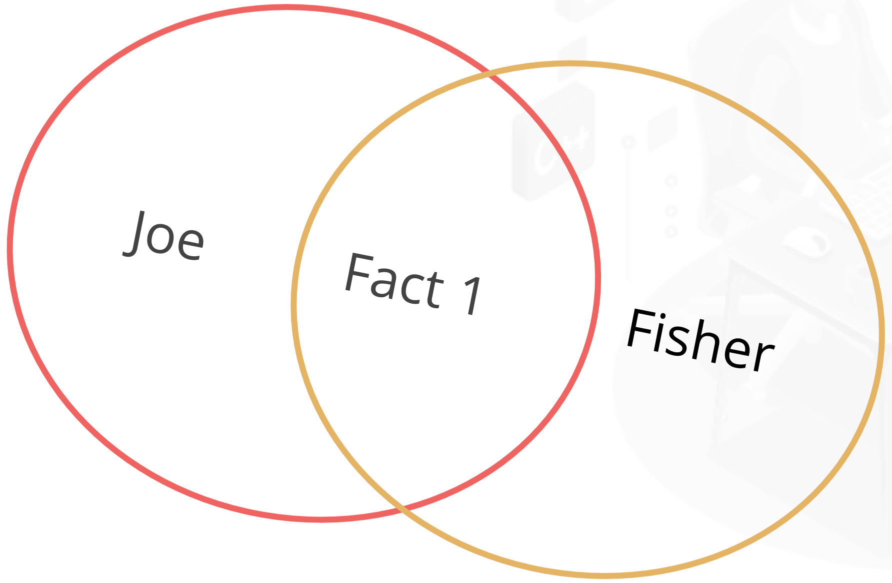
Reference: *https://docs.corda.net/docs/corda-os/4.6/key-concepts-ledger.html*

States

States are immutable digital documents which record the existence, content, and current state of a contract between two or more parties. They are intended to be shared only with those who have an approval to see them.

Fact 1

Seller	Buyer	Price (Rs.)	Due Date	Paid Price	Remaining Price
Joe	Fisher	20,000	June 2022	5000	15,000



States

Historic State

Seller	Buyer	Price (Rs.)	Due Date	Paid Price	Remaining Price
Joe	Fisher	20,000	June 2022	5000	15,000

↓

Current State

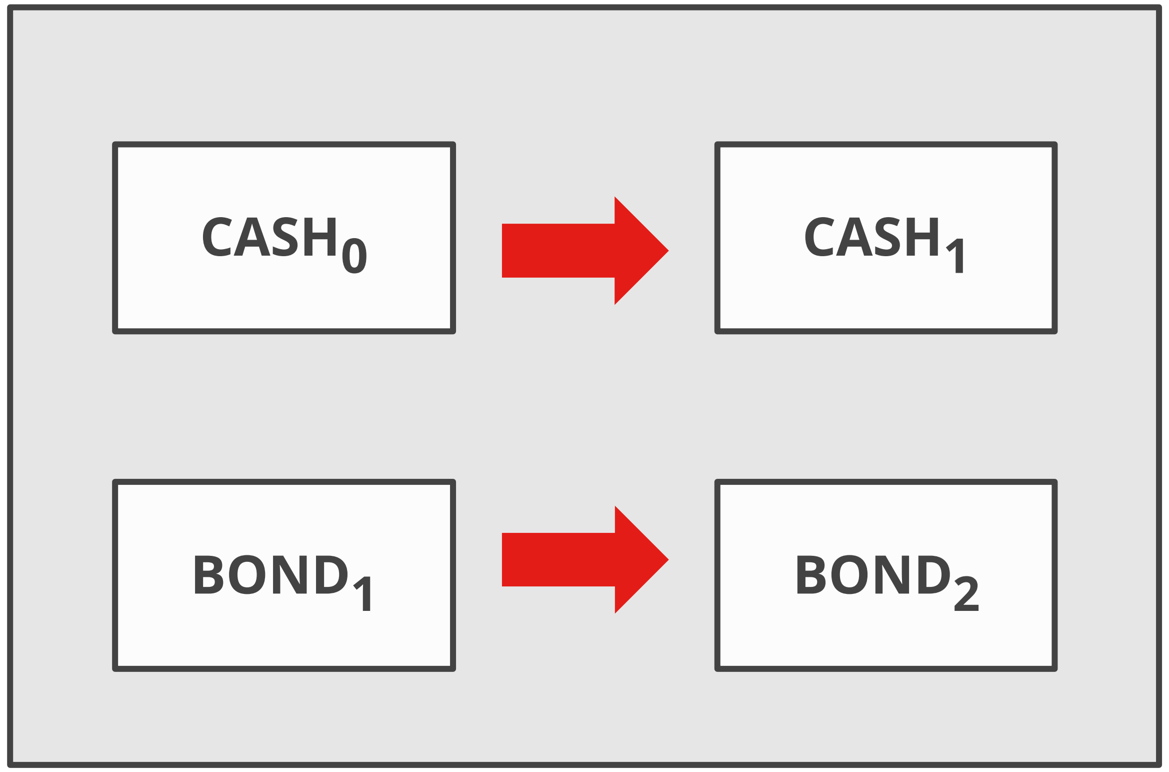
Seller	Buyer	Price (Rs.)	Due Date	Paid Price	Remaining Price
Joe	Fisher	20,000	June 2022	20,000	

- After changing the state, a new state defined as the current state is created. The altered state becomes the historic state.
- Historic states are not stored on the ledger, but they are still available, ensuring that they are never removed.

Transactions

Transactions are generated wherever there is any output for given input. These transactions ultimately get stored in ledger.

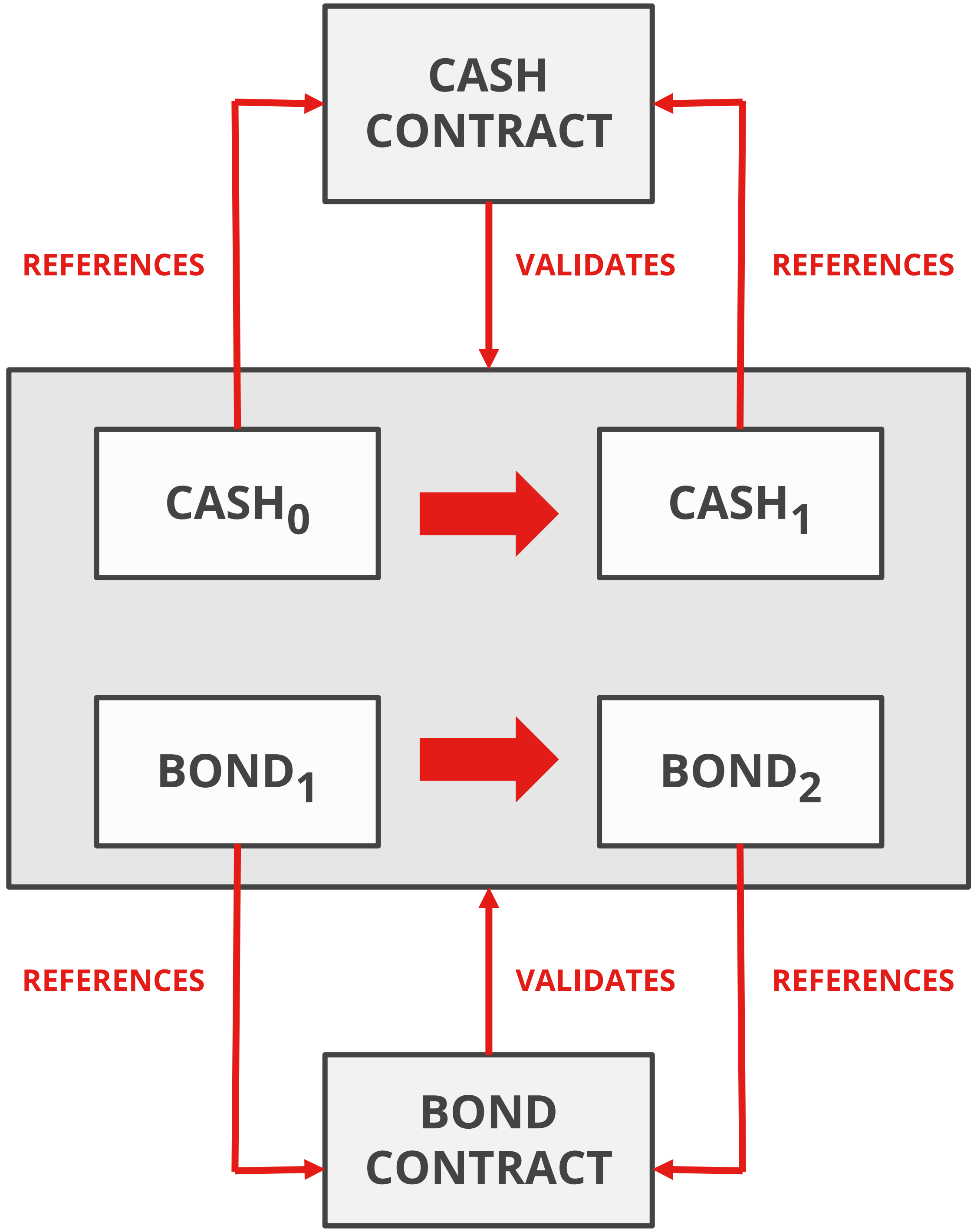
TRANSACTION



CASH$_0$ → CASH$_1$

BOND$_1$ → BOND$_2$

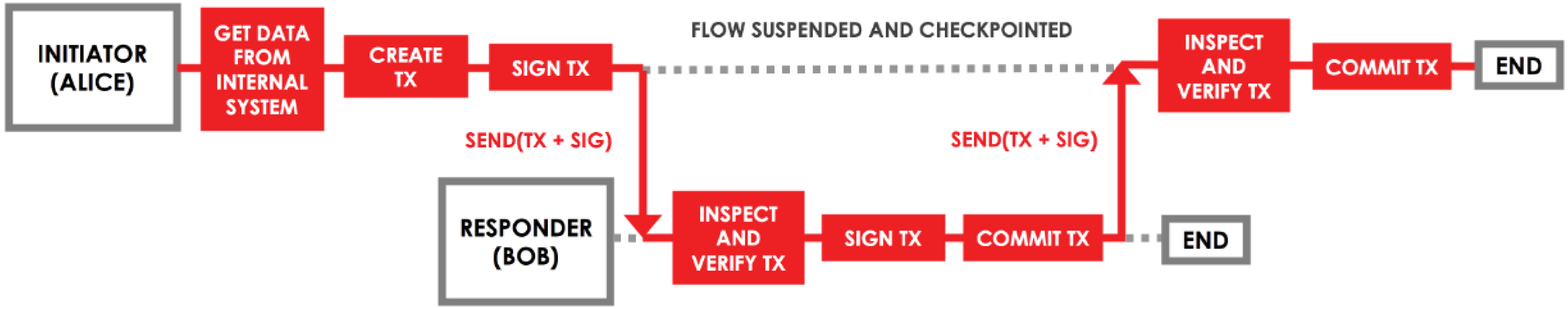
Contracts



Contracts are digital agreements between two or more parties. They are used to verify the transaction is happening as per the defined rules.

Flows

A flow is a method for two or more parties to agree on a simple update ledger and it's usually point-to-point.

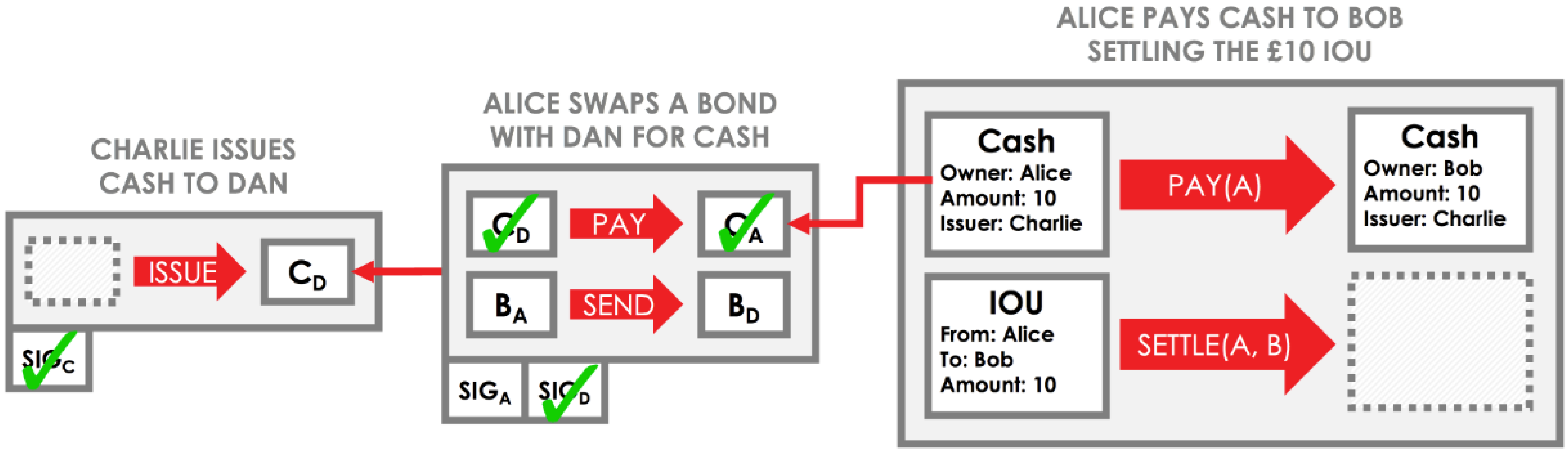


Reference: *https://docs.corda.net/docs/corda-os/4.6/key-concepts-flows.html*

Validity Consensus

Validity consensus is the process of checking that the following conditions hold for every transaction in the transaction chain, which generates the inputs to the proposed transaction:

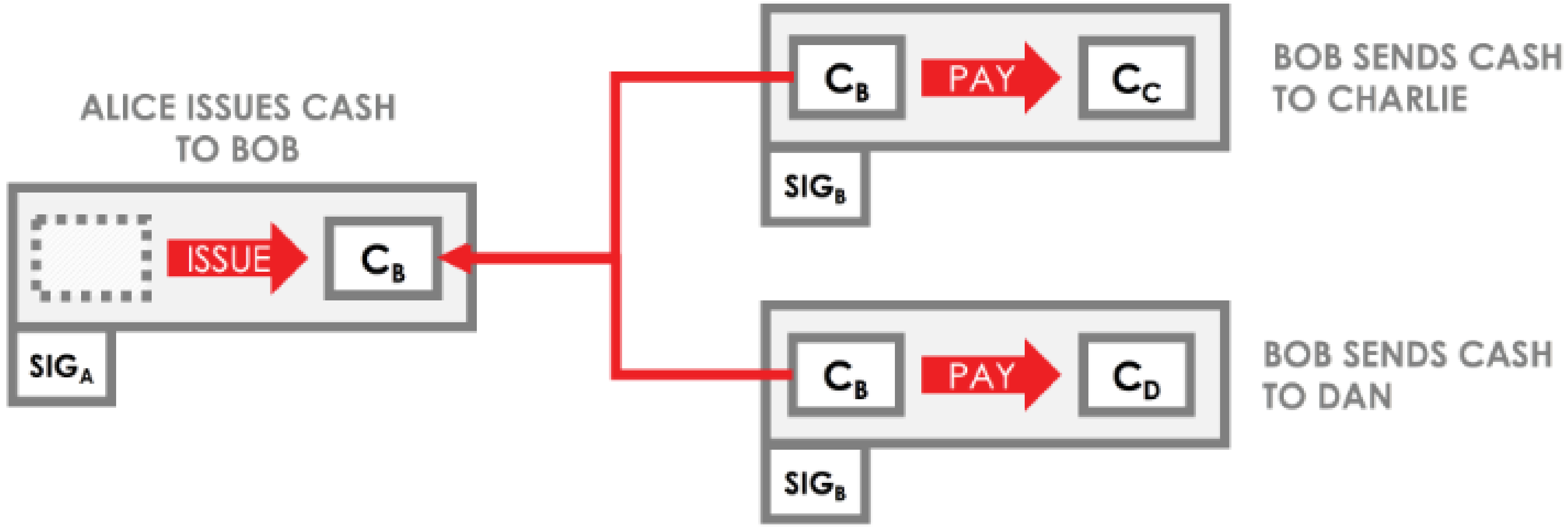
- The transaction is accepted by the contracts of every input and output state
- The transaction has all the required signatures



Reference: *https://docs.corda.net/docs/corda-os/4.6/key-concepts-consensus.html*

Uniqueness Consensus

- The criterion of uniqueness consensus is that none of the inputs to a proposed transaction are being used in another transaction.
- A double spend exists, if one or more of the inputs have already been used in another transaction and the transaction request is deemed invalid.

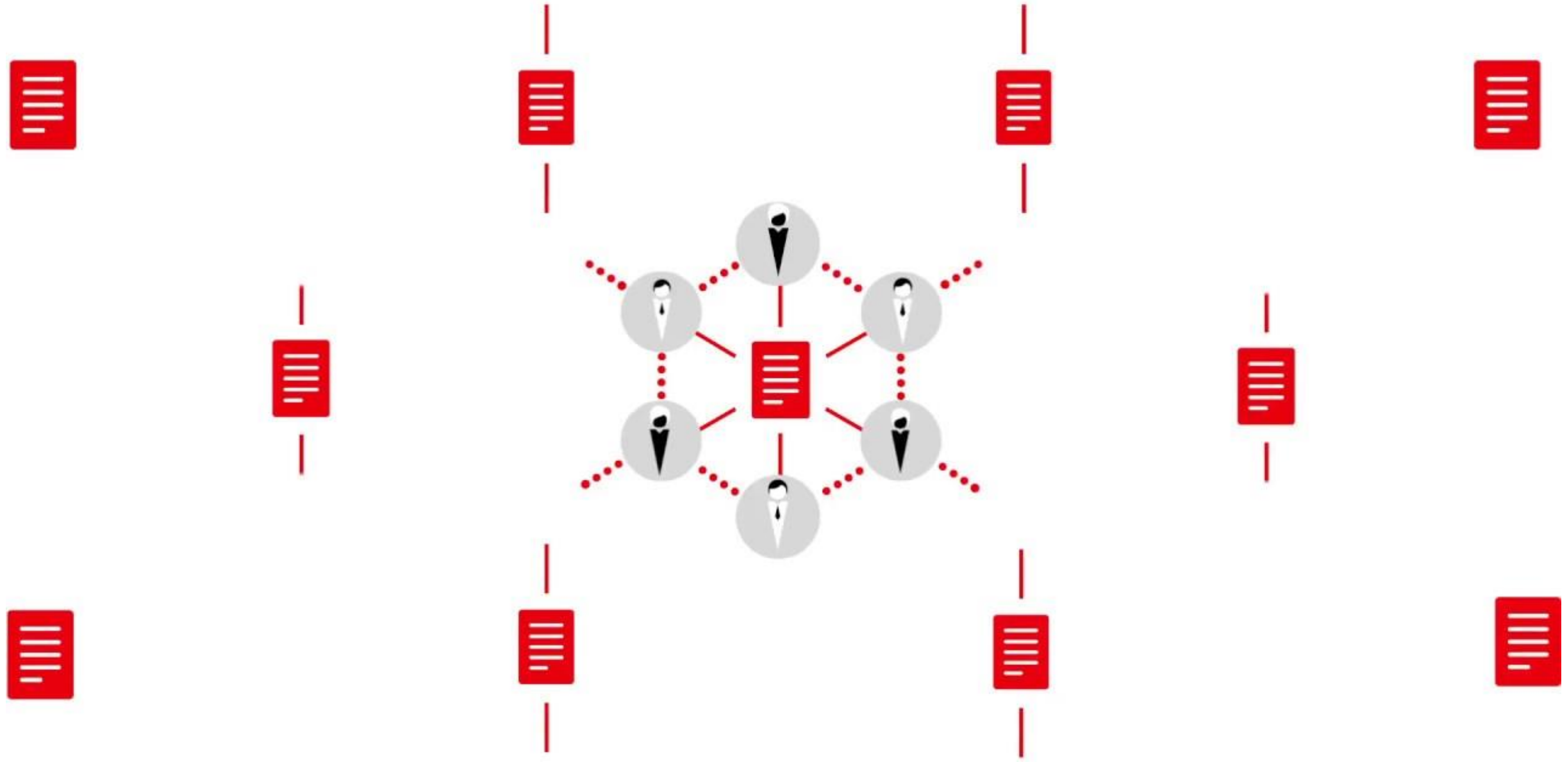


Reference: *https://docs.corda.net/docs/corda-os/4.6/key-concepts-consensus.html*

Corda Network

Corda Network

Corda network is a peer-to-peer network where each node runs instance of Corda. Communication between nodes inside network is secured and point-to-point.

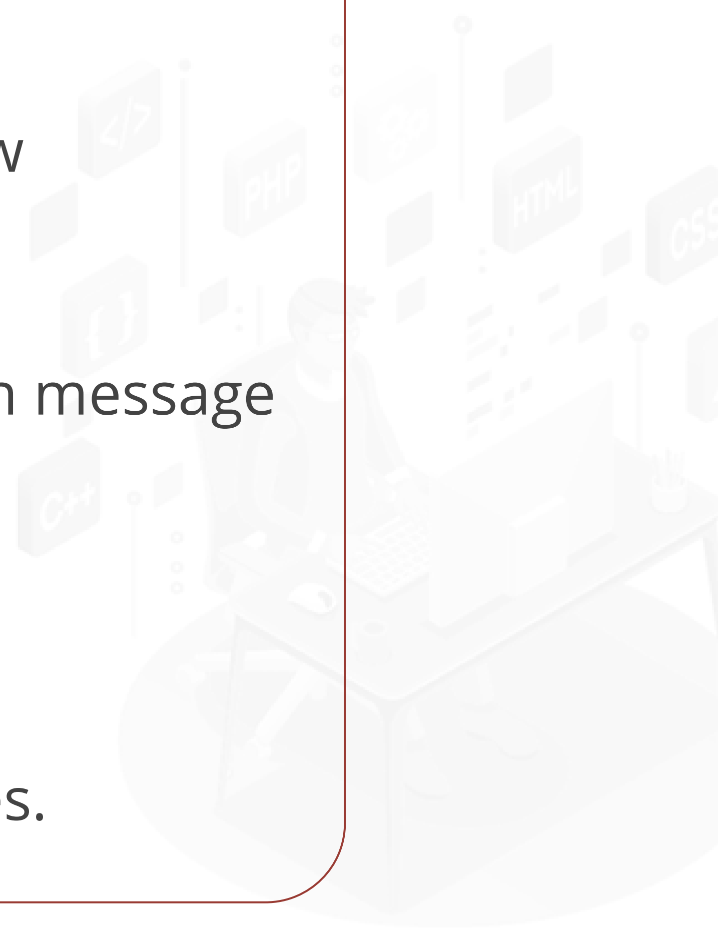


Corda Network Setup

Corda Network Services

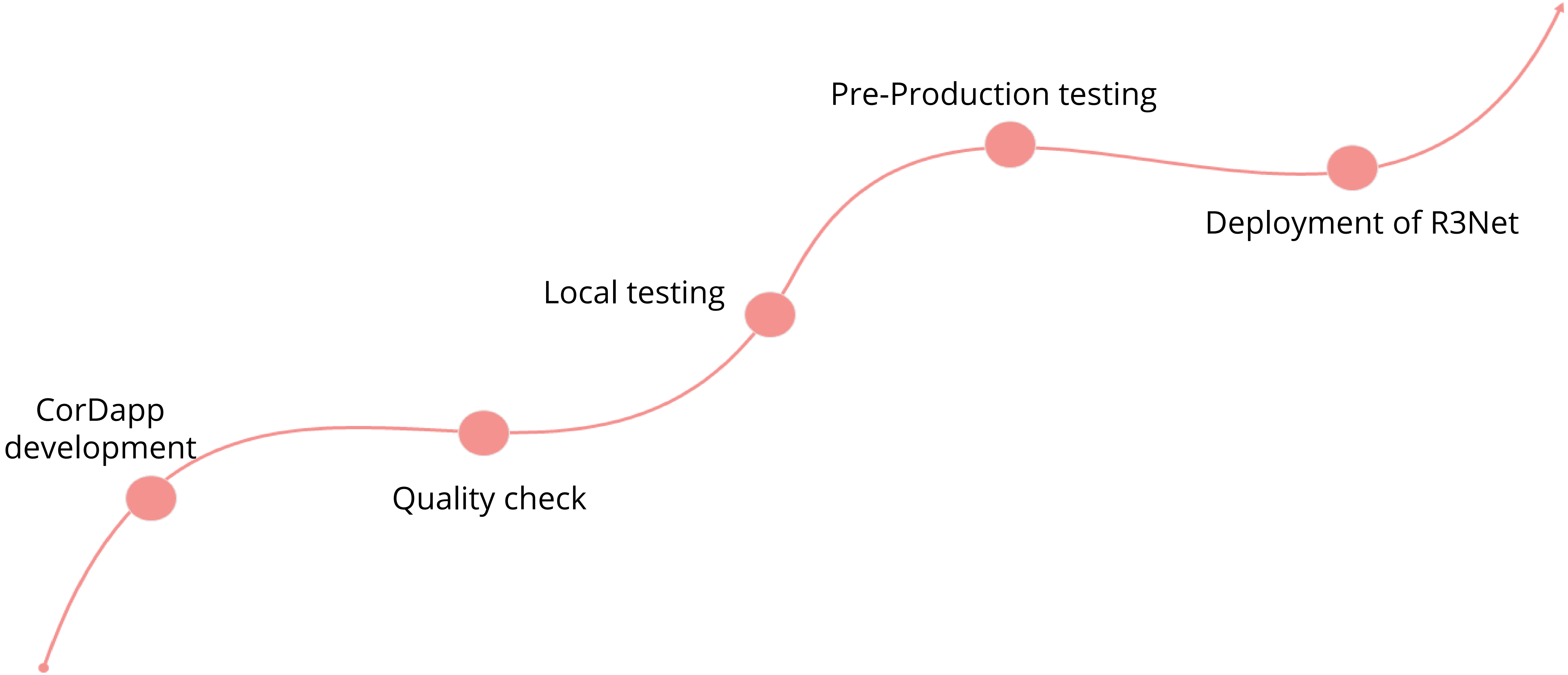
Corda network has mainly four services:

- **Identity Service:** This service handles network node identity. It helps to add new participants to the network by verifying their certificate signing request.
- **Network Map Service:** This service assists in network mapping where it helps in message routing and connecting the nodes with each other.
- **Notary Service:** This service helps in uniqueness consensus.
- **Support Service:** This helps to resolve inquiries related to the above three nodes.



Corda Pre-Production Network

Corda pre-production or User Acceptance Testing is a network that allows developers and businesses to test their CorDapps in a production-like environment.



Key Takeaways

- Enterprise Blockchain is permissioned Blockchain that streamlines the business processes extensively
- Hyperledger Umbrella Project comprises tools, services, and libraries to build Enterprise Blockchain applications
- Hyperledger Fabric is a Blockchain platform that supports general purpose programming languages
- Corda is a distributed ledger platform that uses smart contracts and reduces transaction and record-keeping cost

Lesson-End Project

Transform the Supply Chain

The traditional seafood supply chain industry has illegal, unreported, and unregulated fishing practices. You are required to bring traceability and accountability to the supply chain through the power of Hyperledger Sawtooth technology. Perform the following steps:

1. Visit https://demo.bitwise.io/fish/
2. Click **SignUp** to create an agent
3. Click **Add Fish** to add a fish, which is to be tracked
4. Enter the details of the fish
5. Once the fish is added to the network, you can account and track the supply chain

LESSON-END PROJECT