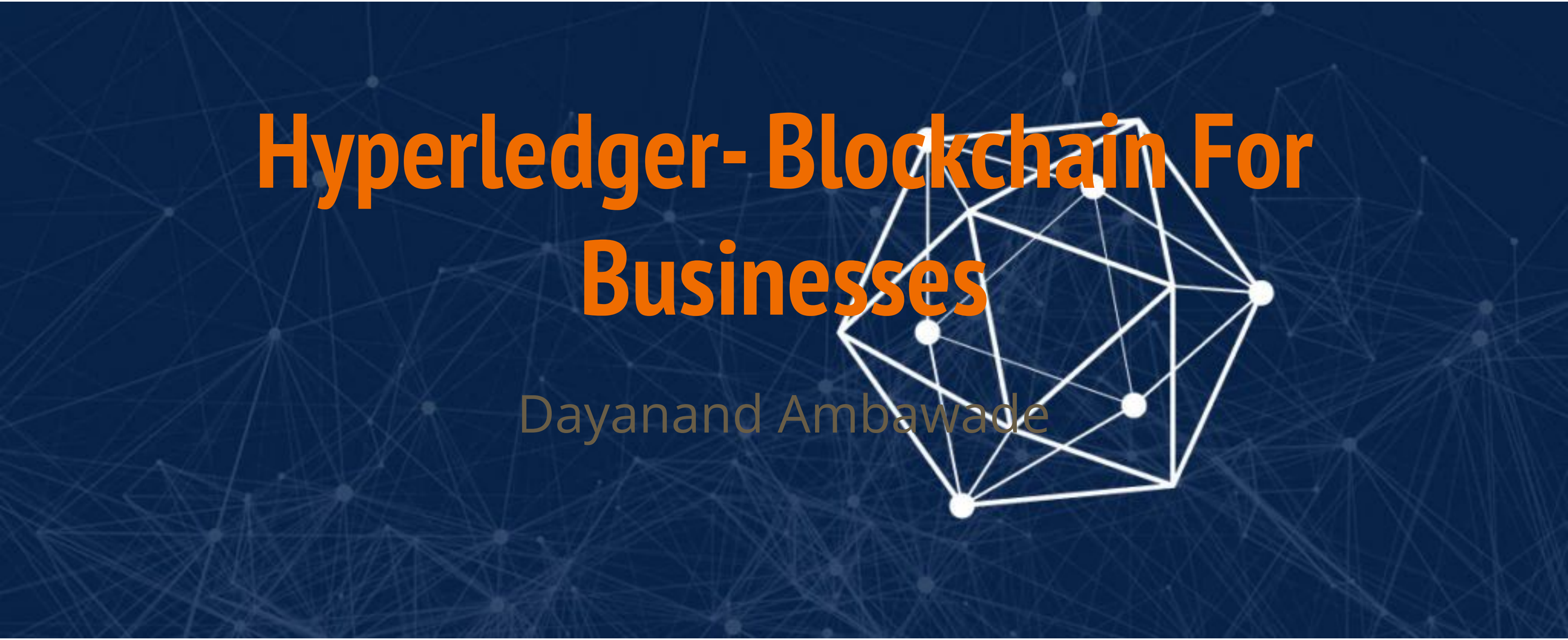


Hyperledger- Blockchain For Businesses

Dayanand Ambawade

Outline:

- Hyperledger overview
- Blockchain as a service
- Architecture and core components
- Hyperledger Fabric model
- Bitcoin versus Ethereum versus Hyperledger
- Hyperledger Fabric capabilities
- Lab using the Tuna application

HYPERLEDGER
BLOCKCHAIN TECHNOLOGIES FOR BUSINESS

Hyperledger overview

Hyperledger is an **open source initiative focused on covering core industry needs with distributed ledger technologies.**

It's a group program **hosted by the Linux Foundation** in collaboration with several industry giants in **information technology, banking, logistics, transportation, finance, manufacturing, and IoT.**

Founded in December 2015, Hyperledger has been appreciated and adopted by several industry leaders such as **Accenture, Airbus, American Express, Cisco, Fujitsu, Hitachi, IBM, Intel, SAP, NEC, BBVA, Bitmark, Bosch, CA Technologies, Capgemini, EY, Factom, H3C, NSE, Oracle, PwC, Redhat, Samsung, Ripple, Thales, Wipro, the Cloud Security Alliance, and many more**.

HYPERLEDGER
BLOCKCHAIN TECHNOLOGIES FOR BUSINESS

Hyperledger- Focus

Hyperledger is the only distributed ledger technology framework that was built to be **granular for businesses** that were in need of **permissioned blockchains to achieve better control over an entire system.**

Hyperledger does **not support** any **cryptocurrency platform** or related system, as it is more about solving critical business problems.

The Hyperledger project was also planned for collaboration between every **blockchain enthusiast, blockchain communities, corporates, and nonprofit organizations** with a single and comprehensive standard of building distributed ledger applications.

E.g Wordpress for website

Blockchain-as-a-service (BaaS)

Since the **birth of cloud computing, o**ne of the hottest terms that has changed the way a product or a service can be delivered or deployed is X (anything) as a service, where X is any form of software or application.

After the **world recognized the immersive power of blockchain, industry leaders began to explore various possibilities of using blockchain with their existing cloud infrastructure models such as supply chain management, identity and access control, database management, and many more.**

Hyperledger resembles to the distributed ledger technology however blockchain technology has been taken a special focus in the ecosystem.

With the **Azure Blockchain** service, Microsoft became the first software vendor to launch BaaS in 2015. Microsoft, in close collaboration with **ConsenSys,** announced that it was going to develop an Ethereum BaaS on the Microsoft Azure platform. SAP launched its own BaaS platform and named it **Leonardo,** which is a Hyperledger-based cloud service.

Deloitte, the largest consulting firm, has come up with a blockchain-based business solution and named it **Rubix Core**. It is an architecture designed for building private and customized networks for their clients.

Hyperledger Program goal

It is an upfront effort to develop cross-industry frameworks for platform collaboration.

The financial industry has been the most active in collaborating with Hyperledger platforms to achieve a seamless move.

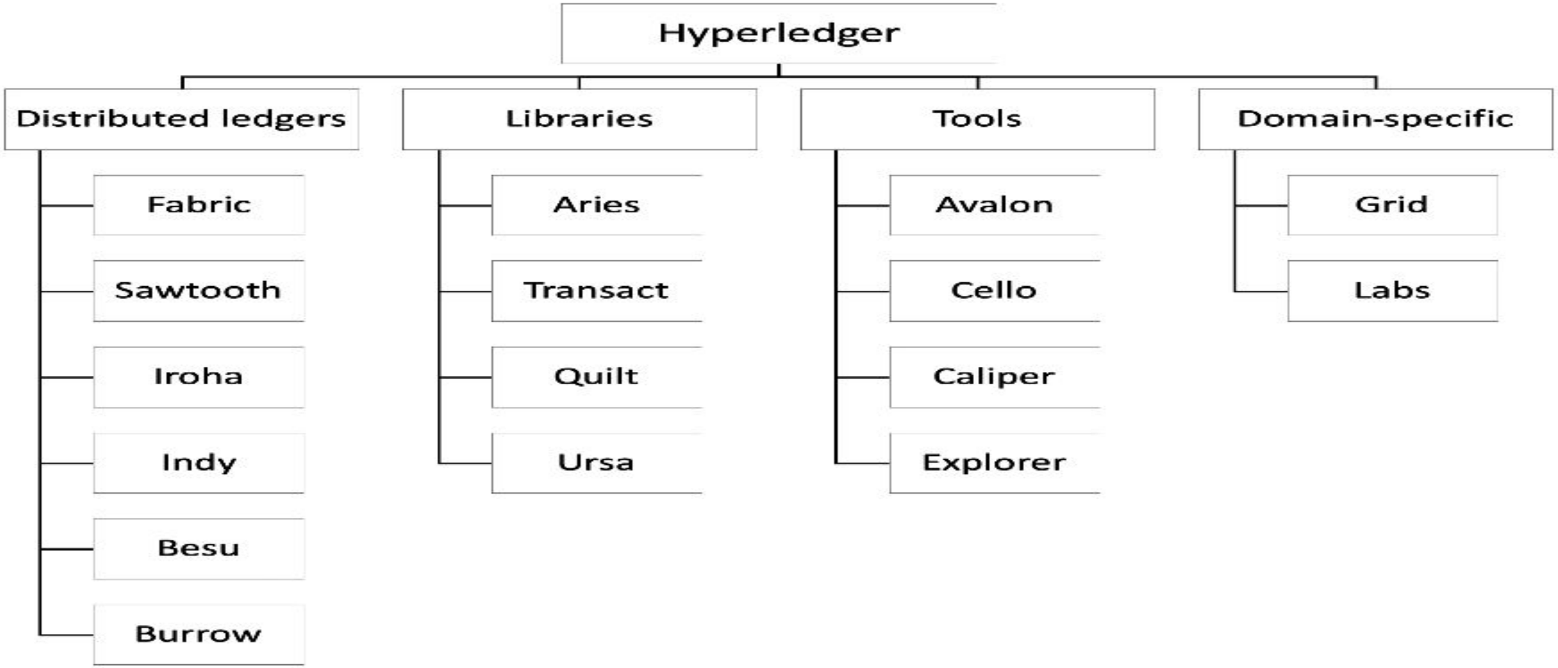
Community-driven infrastructure: As the Hyperledger project is supported by several private and government institutions, it presents a highly efficient and open community-driven environment

Enterprise-grade framework: Unlike the cryptocurrency blockchain, Hyperledger was developed to support businesses to perform secure and reliable transactions over distributed ledger networks

Building technical communities: The project is also aimed at building a more effective and larger technical community to innovate and develop blockchain smart contacts and other related code

Awareness: It's a great way to spread awareness to businesses and other institutions about blockchain technology and its business use cases.

Hyperledger Project



Architecture and core components

Shared ledger: It is an append-only ledger, and it stores the blocks in chronological order

Consensus algorithm: It's a method to achieve a common agreement over a change in the distributed ledger

Privacy: The main purpose of building the Hyperledger was to achieve a permissioned network for secure and reliable transactions in mission-critical business environments

Smart contract: This is a granular method to plan and process transaction requests

Hyperledger Architecture:

Consensus layer: mainly responsible for generating an agreement on each order and validating transactions based on a predefined set of rules

Smart contract layer: takes care of transaction requests and applying business logic

Communication layer: facilitates a platform that allows nodes to communicate over peer-to-peer transport

Data store abstraction: allows various data sources to be used by other modules

Crypto abstraction: allows different crypto algorithms to be used without impacting other modules

Identity service: enables the deployment of root of trust during blockchain setup with additional authentication and authorization cover

Policy service: responsible for managing several policies, such as the consensus policy, endorsement policy, and the group management policy

API: This enables clients and applications to talk to blockchain modules

Interoperation: This provides interoperability among different blockchain instances

Hyperledger Architecture

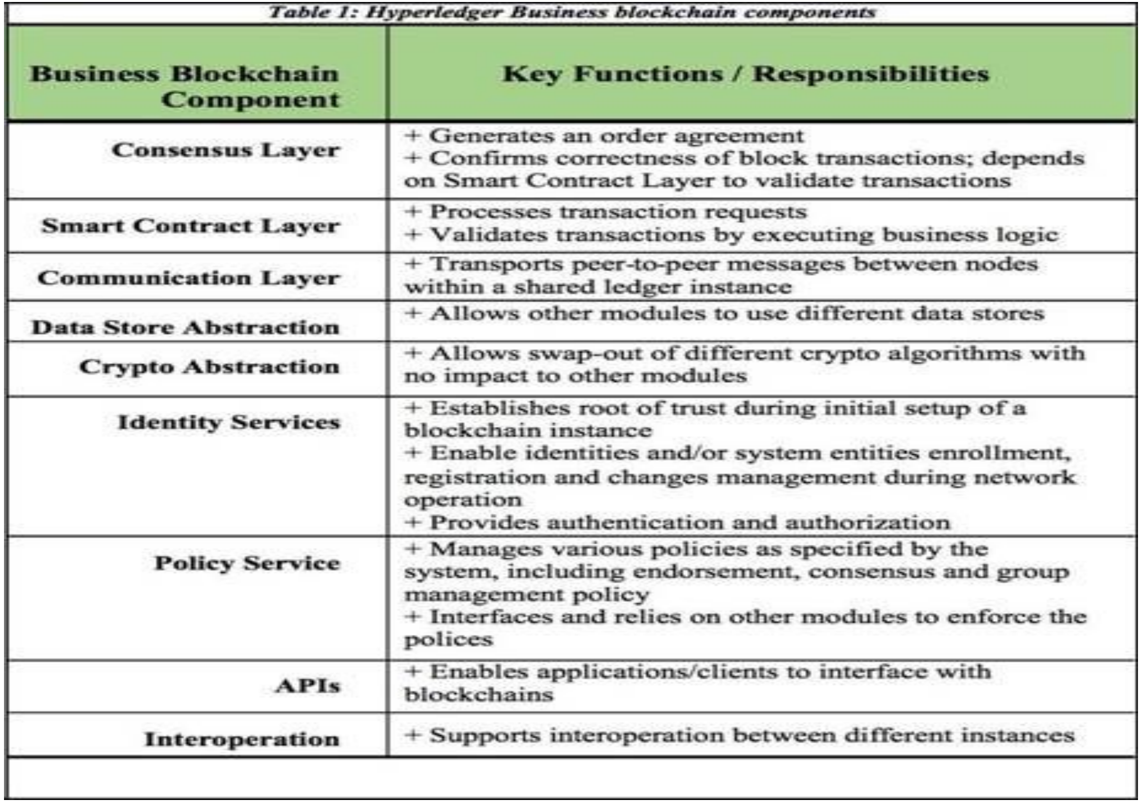
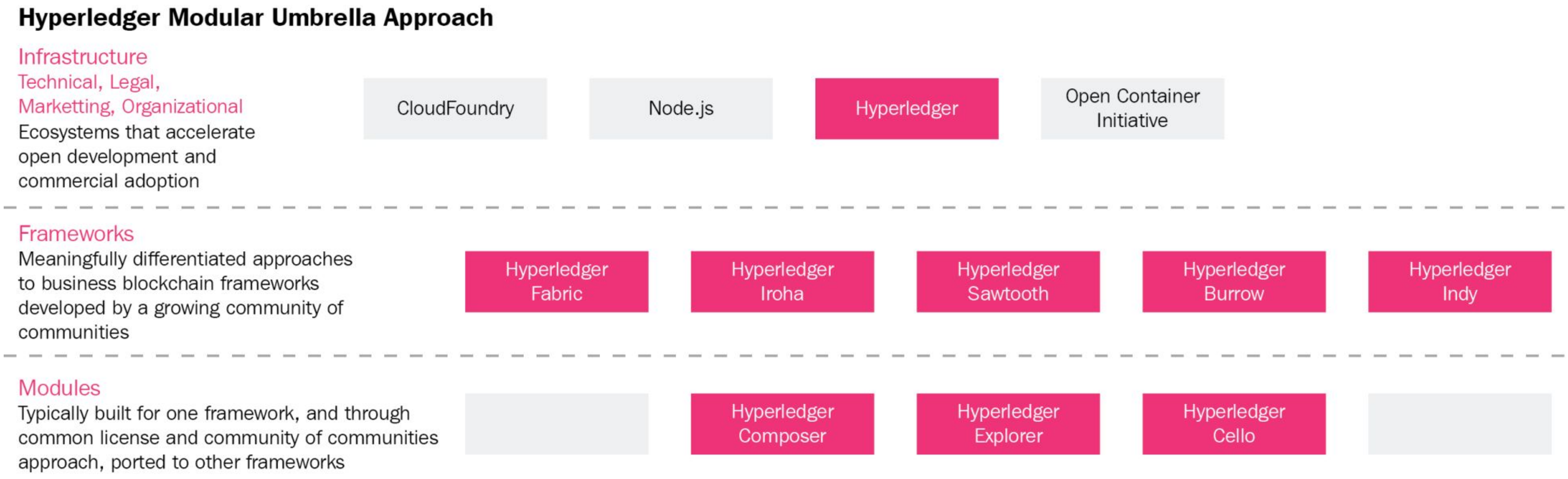


Table 1: Hyperledger Business blockchain components

Business Blockchain Component	Key Functions / Responsibilities
Consensus Layer	+ Generates an order agreement + Confirms correctness of block transactions; depends on Smart Contract Layer to validate transactions
Smart Contract Layer	+ Processes transaction requests + Validates transactions by executing business logic
Communication Layer	+ Transports peer-to-peer messages between nodes within a shared ledger instance
Data Store Abstraction	+ Allows other modules to use different data stores
Crypto Abstraction	+ Allows swap-out of different crypto algorithms with no impact to other modules
Identity Services	+ Establishes root of trust during initial setup of a blockchain instance + Enable identities and/or system entities enrollment, registration and changes management during network operation + Provides authentication and authorization
Policy Service	+ Manages various policies as specified by the system, including endorsement, consensus and group management policy + Interfaces and relies on other modules to enforce the polices
APIs	+ Enables applications/clients to interface with blockchains
Interoperation	+ Supports interoperation between different instances

Hyperledger framework



Hyperledger Modular Umbrella Approach

Infrastructure
Technical, Legal,
Marketting, Organizational
Ecosystems that accelerate open development and commercial adoption

CloudFoundry | Node.js | Hyperledger | Open Container Initiative

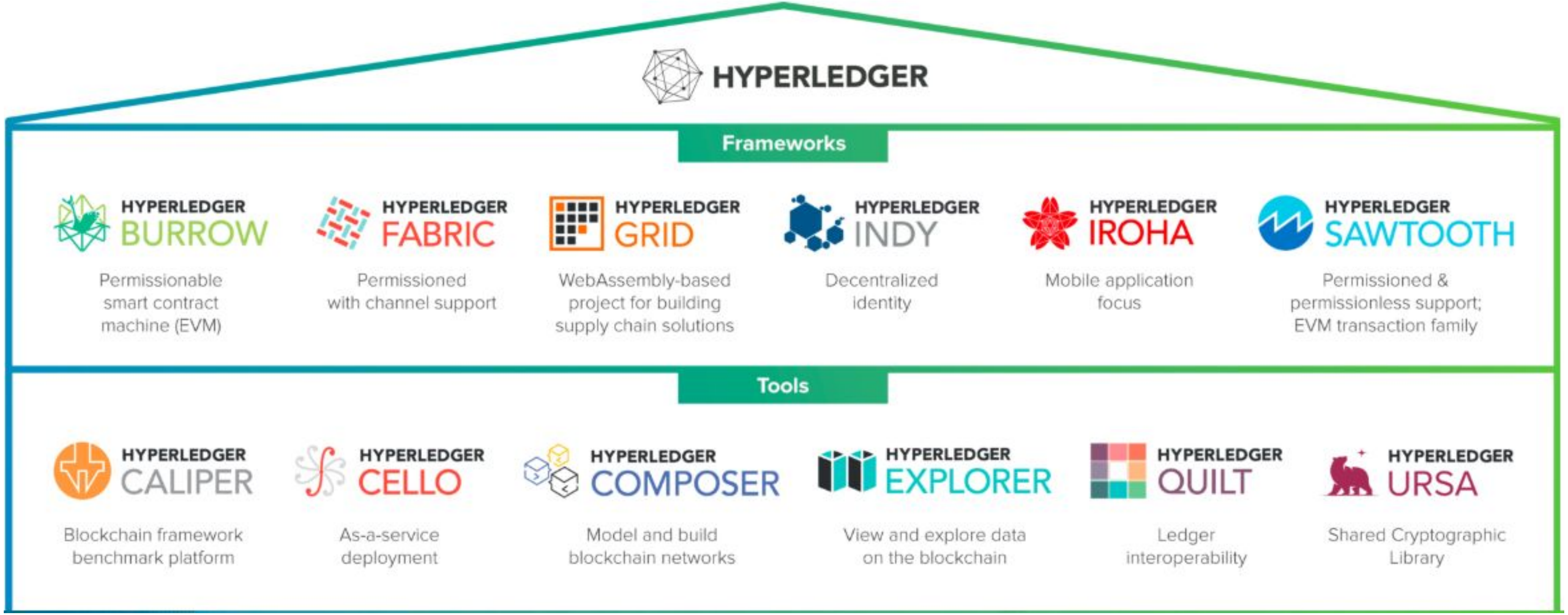
Frameworks
Meaningfully differentiated approaches to business blockchain frameworks developed by a growing community of communities

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

Modules
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 Frameworks



HYPERLEDGER

Frameworks

HYPERLEDGER BURROW
Permissionable smart contract machine (EVM)

HYPERLEDGER FABRIC
Permissioned with channel support

HYPERLEDGER GRID
WebAssembly-based project for building supply chain solutions

HYPERLEDGER INDY
Decentralized identity

HYPERLEDGER IROHA
Mobile application focus

HYPERLEDGER SAWTOOTH
Permissioned & permissionless support; EVM transaction family

Tools

HYPERLEDGER CALIPER
Blockchain framework benchmark platform

HYPERLEDGER CELLO
As-a-service deployment

HYPERLEDGER COMPOSER
Model and build blockchain networks

HYPERLEDGER EXPLORER
View and explore data on the blockchain

HYPERLEDGER QUILT
Ledger interoperability

HYPERLEDGER URSA
Shared Cryptographic Library

Hyperledger Nodes

Clients: Clients propose the transaction request on the network. It has to be connected to a peer to participate in blockchain.

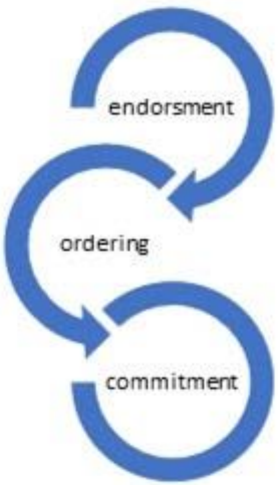
Peers: Peers listen to the ledger update and keep a copy of it. Based on their nature, there could be two further types:

Endorsing peers: Endorsers simulate and endorse transactions

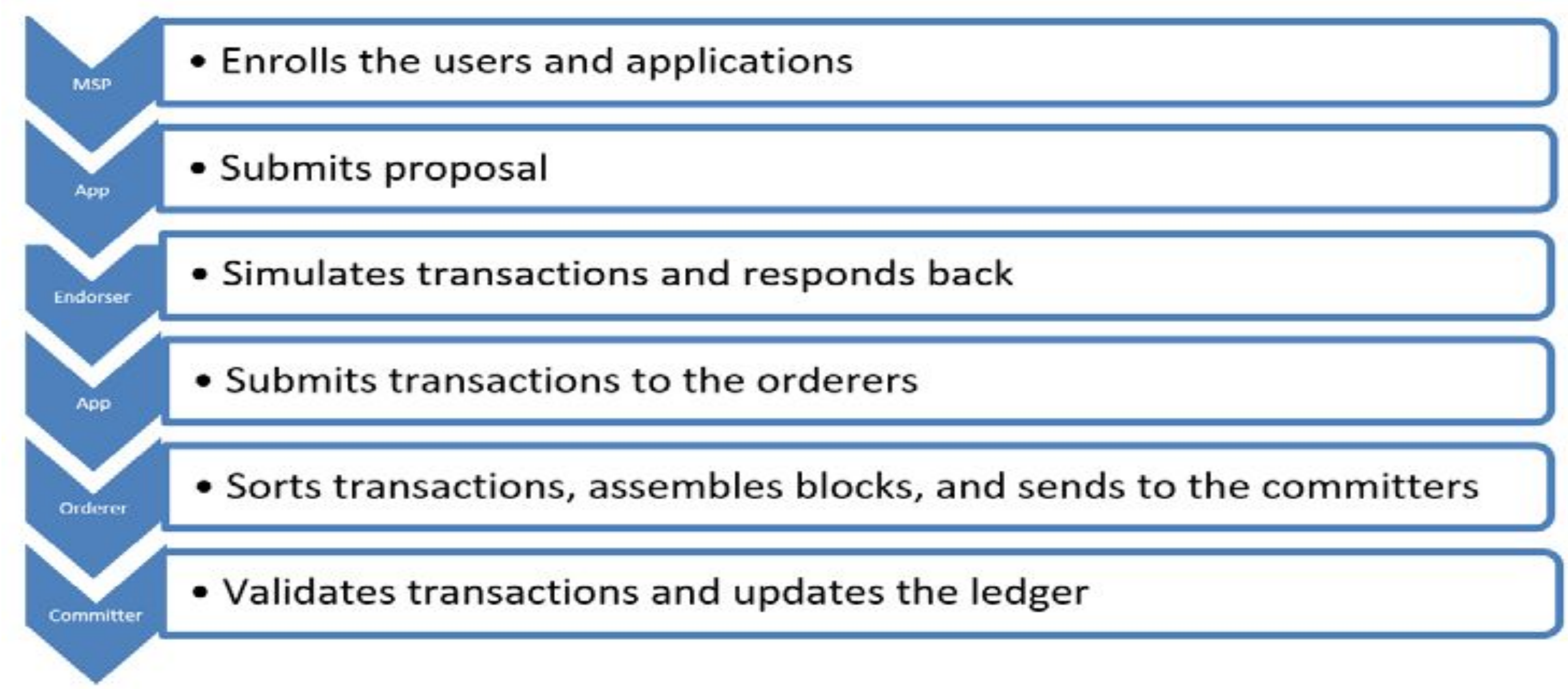
Committing peers: Committers validate transactions before committing transactions in the network

Ordering service: The ordering service accepts endorsed transactions, arranges and orders them into a block, and finally delivers it to committing peers. The ordering service also provides a shared and secure communication channel for clients and peers. It acts as a medium for broadcasting transactions and helps us deliver this to peers.

Hyperledger Nodes



The high-level transaction lifecycle



MSP

- Enrolls the users and applications

App

- Submits proposal

Endorser

- Simulates transactions and responds back

App

- Submits transactions to the orderers

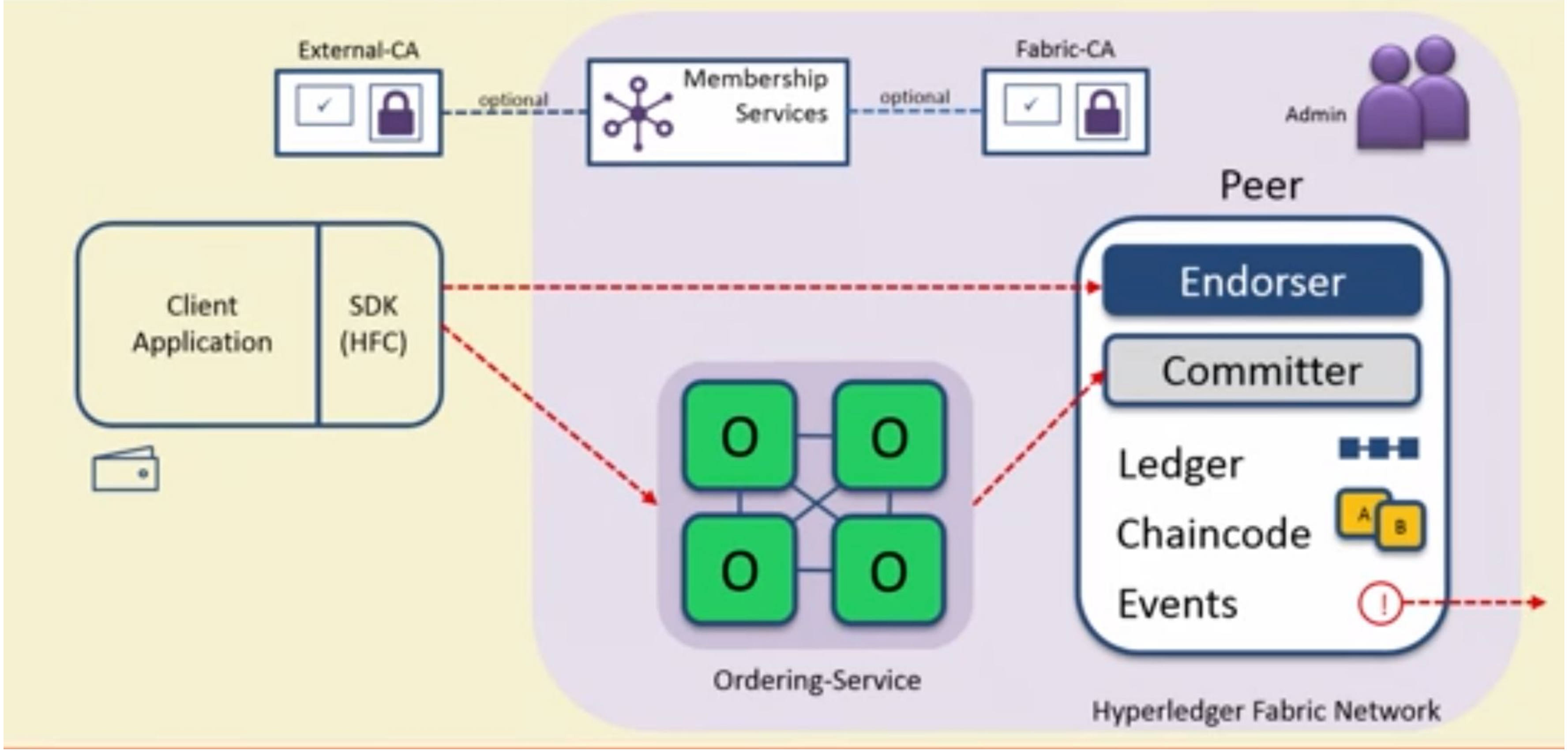
Orderer

- Sorts transactions, assembles blocks, and sends to the committers

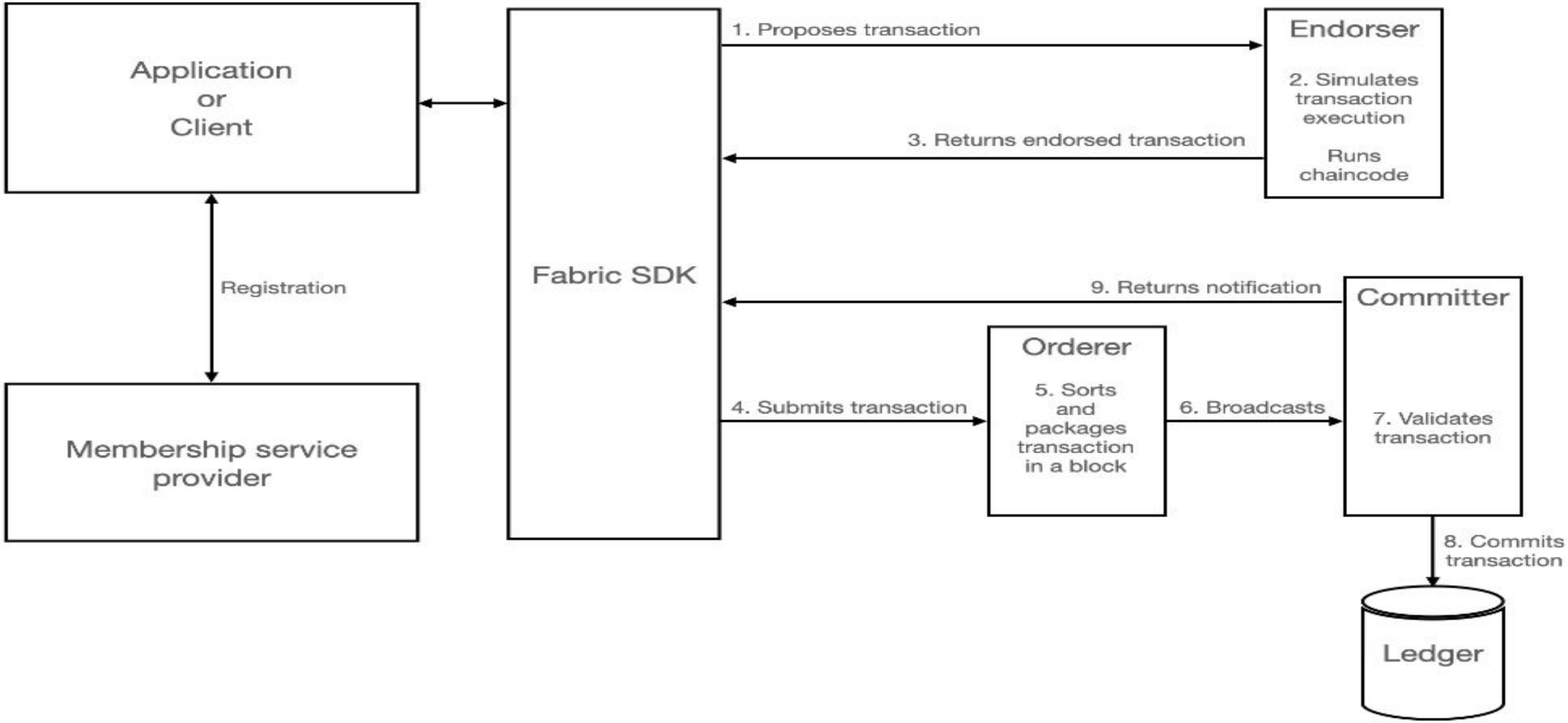
Committer

- Validates transactions and updates the ledger

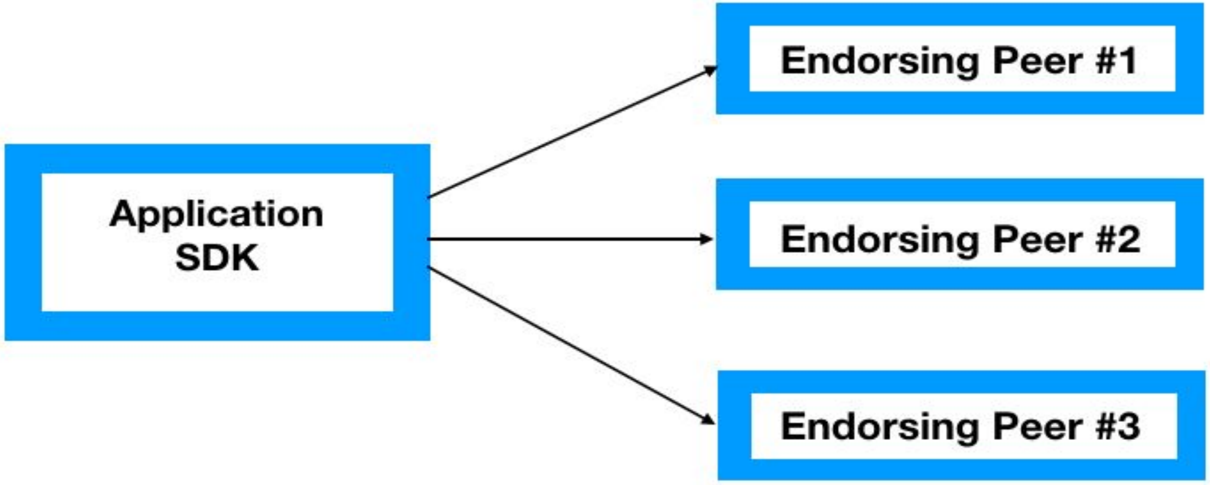
How does Hyperledger work:



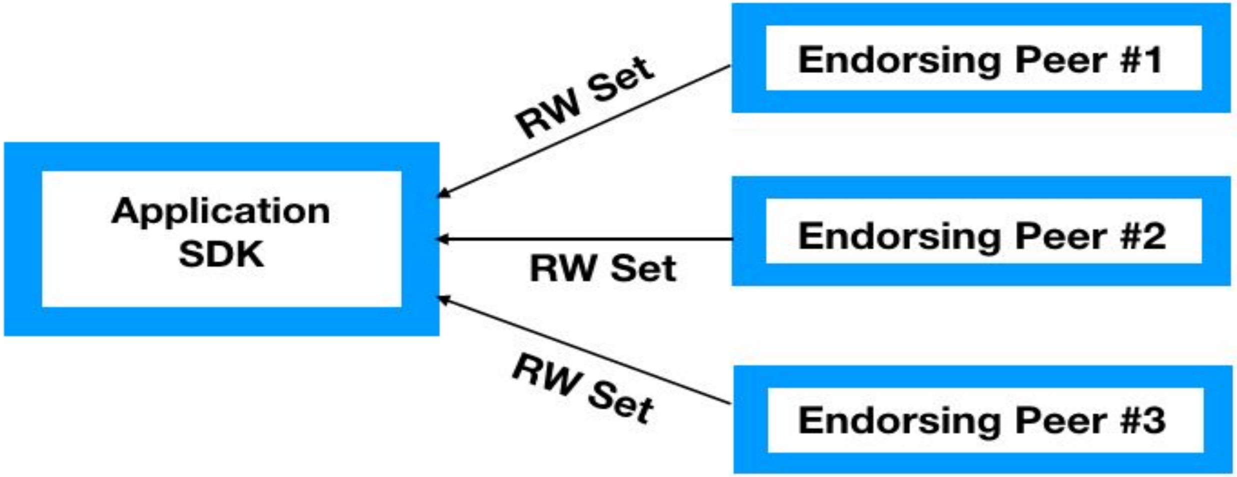
Hyperledger-Transaction Flow



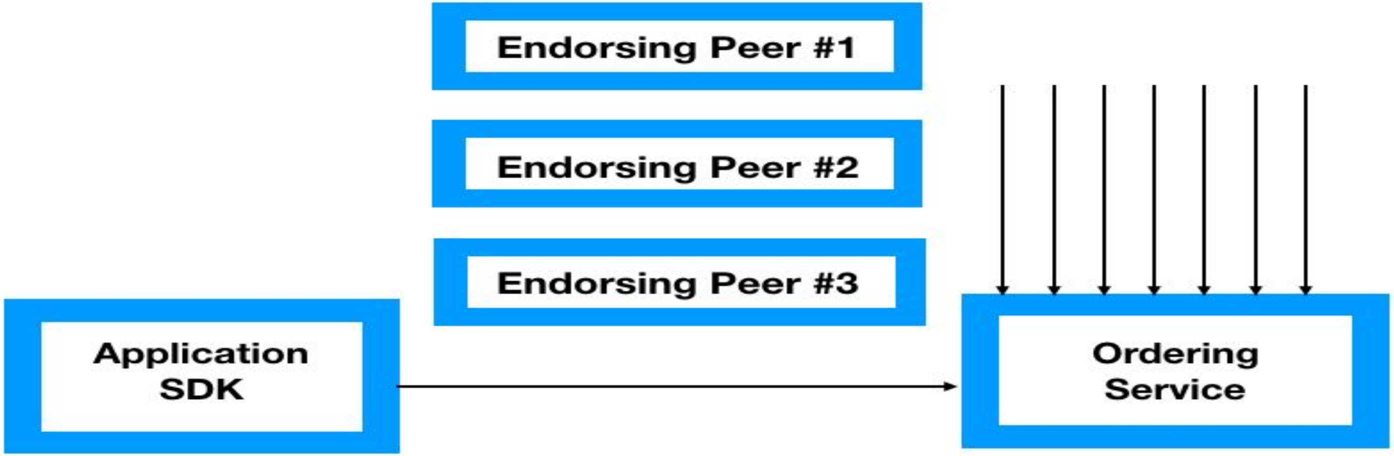
HLF Working



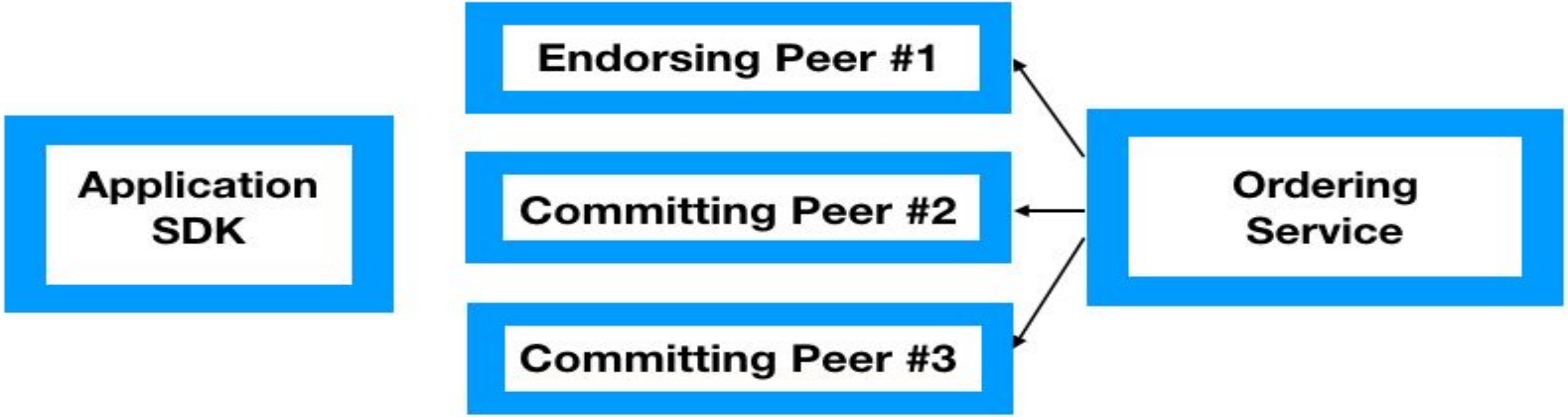
Step1: Transaction Proposal



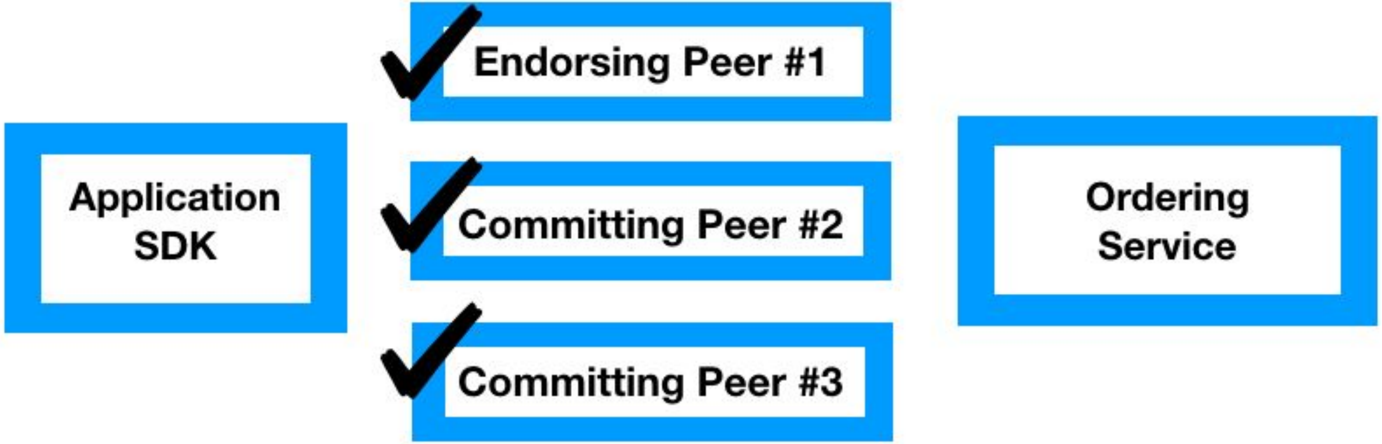
Step2: Endorsers sends RW sets to client



Step3: Client Application Service



Step4: Orderer sends transactions in block to committing peers



Application SDK

✓ Endorsing Peer #1

✓ Committing Peer #2

✓ Committing Peer #3

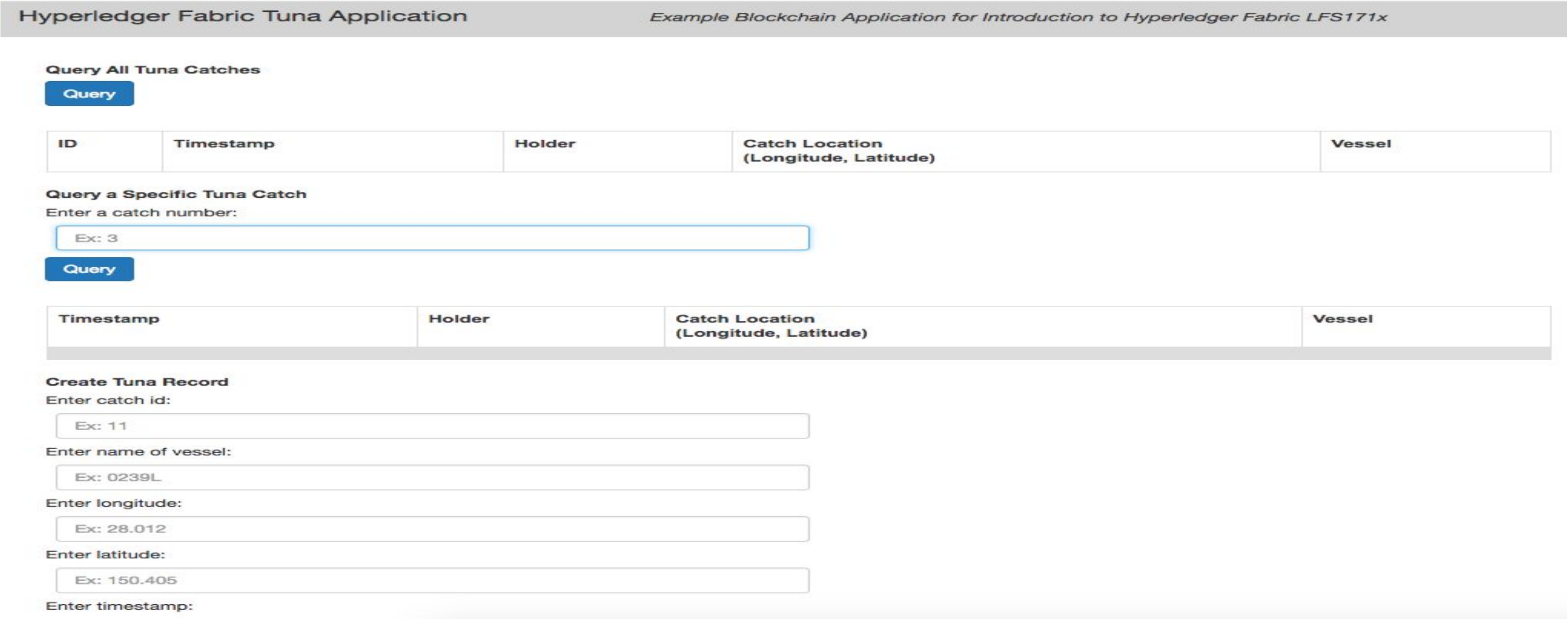
Ordering Service

Step5: Committing peers validate each transaction in the block

Lab on Hyperledger Fabric

Lab8A- Hyperledger Fabric Samples

Lab8B- Tuna Application



Hyperledger Fabric Tuna Application

Example Blockchain Application for Introduction to Hyperledger Fabric LFS171x

Query All Tuna Catches

ID	Timestamp	Holder	Catch Location (Longitude, Latitude)	Vessel

Query a Specific Tuna Catch

Enter a catch number:

Ex: 3

Timestamp	Holder	Catch Location (Longitude, Latitude)	Vessel

Create Tuna Record

Enter catch id:

Ex: 11

Enter name of vessel:

Ex: 0239L

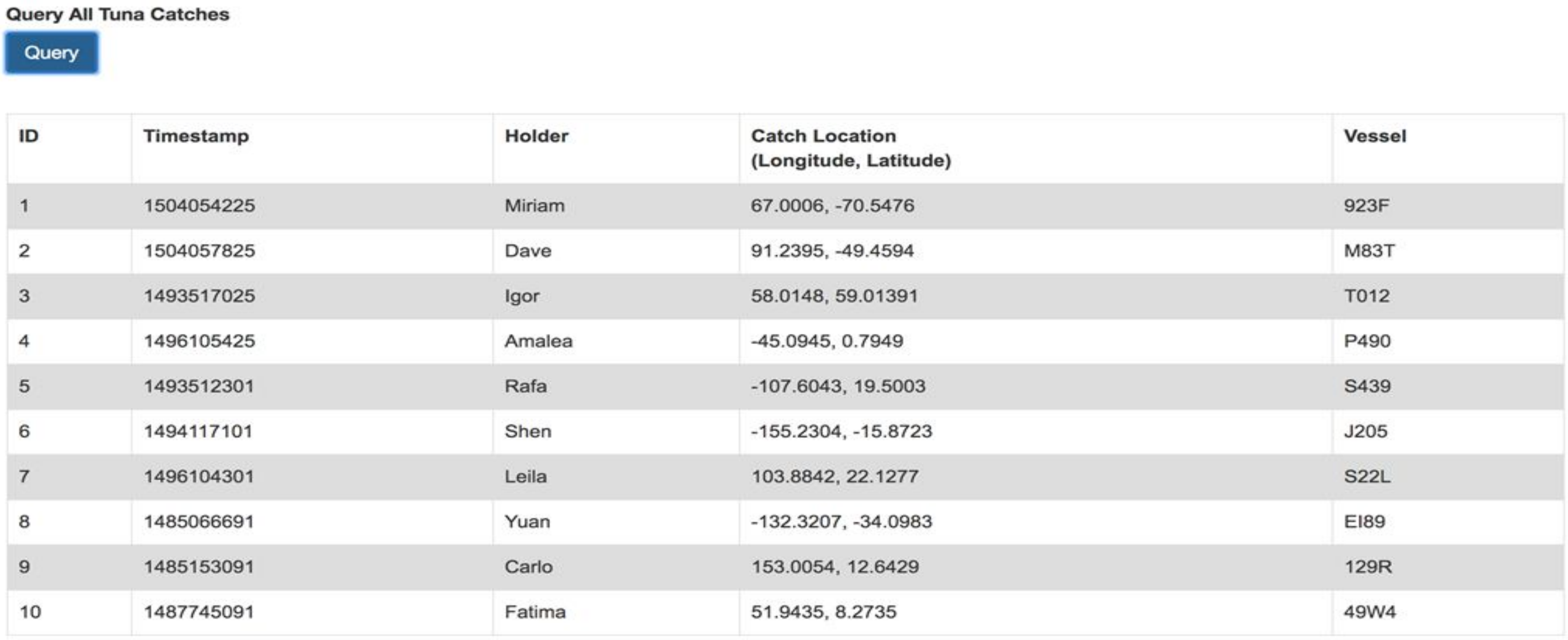
Enter longitude:

Ex: 28.012

Enter latitude:

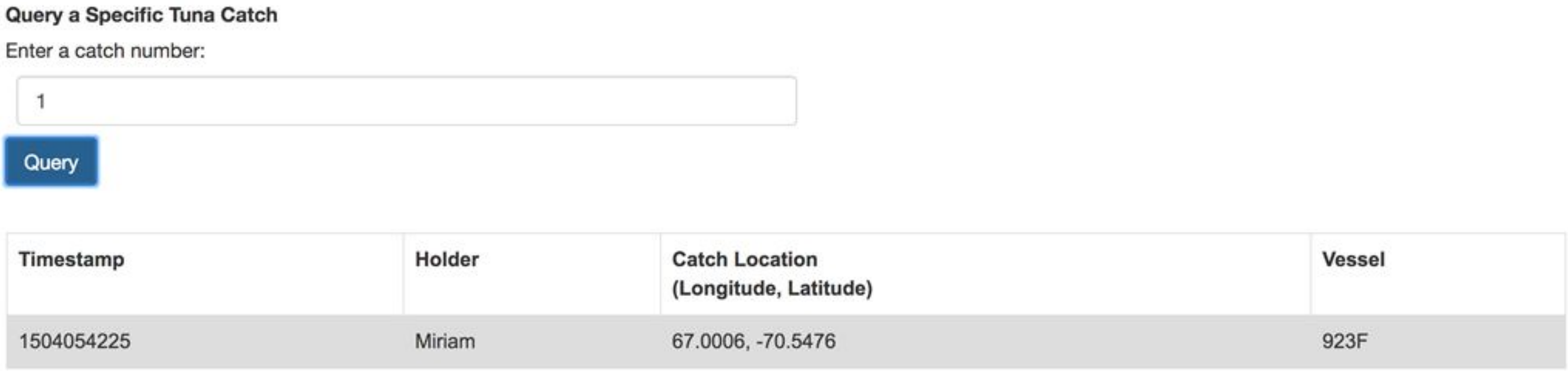
Ex: 150.405

Enter timestamp:



Query All Tuna Catches

ID	Timestamp	Holder	Catch Location (Longitude, Latitude)	Vessel
1	1504054225	Miriam	67.0006, -70.5476	923F
2	1504057825	Dave	91.2395, -49.4594	M83T
3	1493517025	Igor	58.0148, 59.01391	T012
4	1496105425	Amalea	-45.0945, 0.7949	P490
5	1493512301	Rafa	-107.6043, 19.5003	S439
6	1494117101	Shen	-155.2304, -15.8723	J205
7	1496104301	Leila	103.8842, 22.1277	S22L
8	1485066691	Yuan	-132.3207, -34.0983	EI89
9	1485153091	Carlo	153.0054, 12.6429	129R
10	1487745091	Fatima	51.9435, 8.2735	49W4

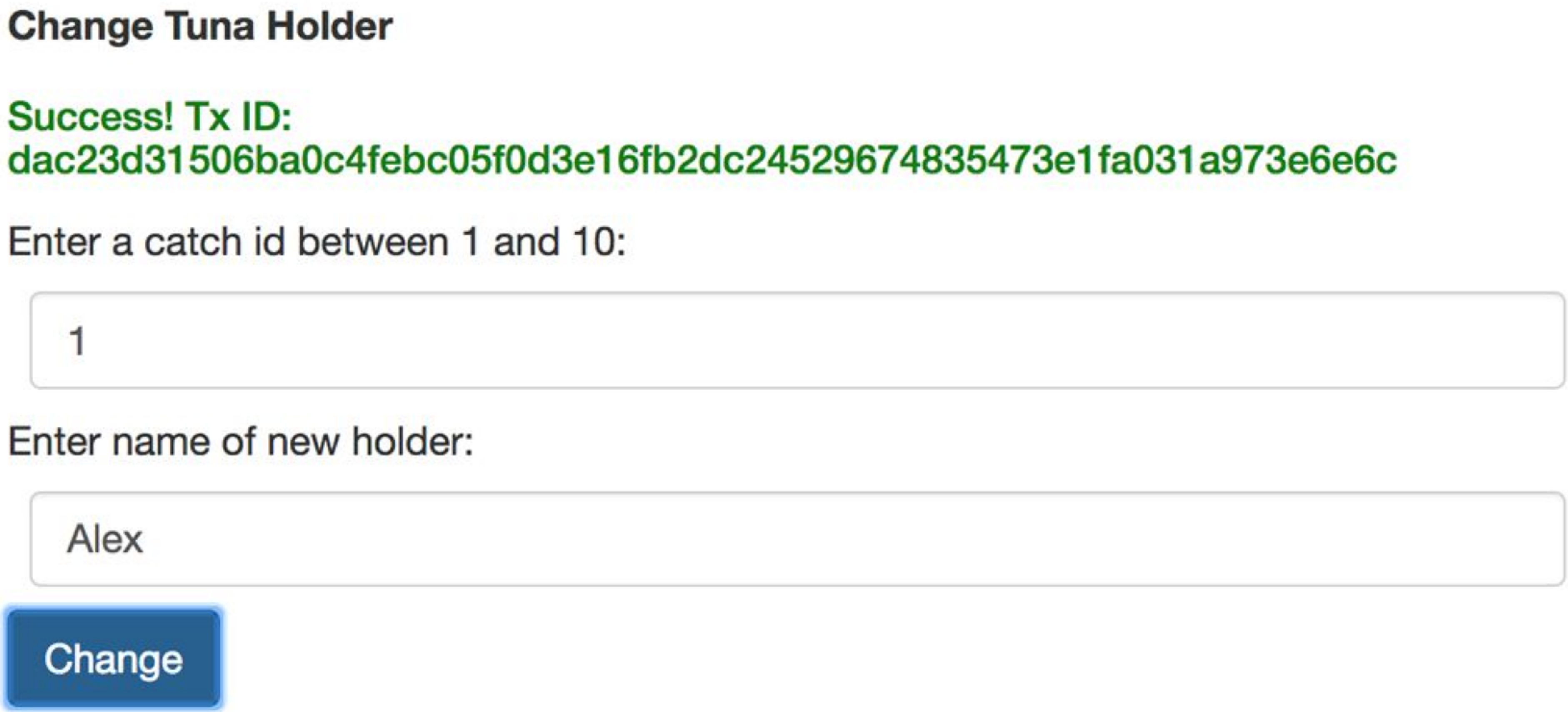


Query a Specific Tuna Catch

Enter a catch number:

1

Timestamp	Holder	Catch Location (Longitude, Latitude)	Vessel
1504054225	Miriam	67.0006, -70.5476	923F



Change Tuna Holder

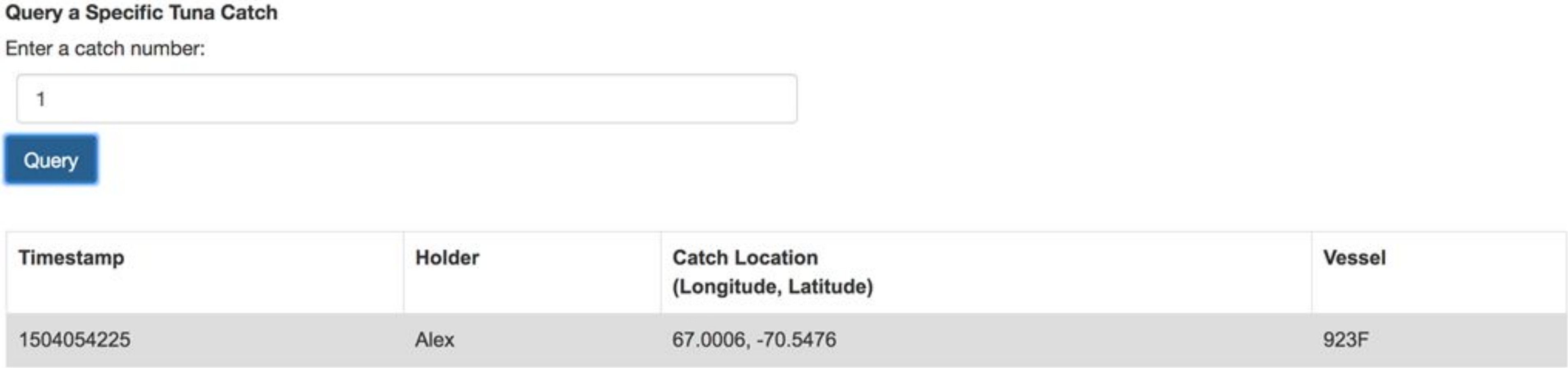
Success! Tx ID:
dac23d31506ba0c4febc05f0d3e16fb2dc24529674835473e1fa031a973e6e6c

Enter a catch id between 1 and 10:

1

Enter name of new holder:

Alex



Query a Specific Tuna Catch

Enter a catch number: 1

Timestamp	Holder	Catch Location (Longitude, Latitude)	Vessel
1504054225	Alex	67.0006, -70.5476	923F

References:

[1] Mastering Blockchain- 4th Edition by Imran Bashir, Packt Publications Chapter-14 Hyperledger

[2]Hyperledger Fabric

Introduction — hyperledger-fabricdocs master documentation

[3] Install Hyperledger Fabric in Ubuntu 20.04 LTS

Install Hyperledger Fabric in Ubuntu 20.04 LTS | by Bibek Poudel

[4] Hyperledger Architecture, Projects, Tools and Libraries | Coding Bootcamps Blockchain Training

[5] https://github.com/hyperledger-archives/education/tree/master/LFS171x