Concepts & Deep Dive

raj@acloudfan.com

@acloudfan

http://ACloudFan.com

Discount Coupon Links to UDEMY courses:

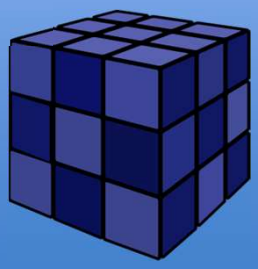
https://www.udemy.com/hyperledger/?couponCode=DKHLF1099

https://www.udemy.com/ethereum-dapp/?couponCode=DKETH1099

https://www.udemy.com/rest-api/?couponCode=DKRST1099

mentoring, seeking Blockchain part time work, project guidance, advice … …
http://www.bcmentors.com

This deck is part of a online course on "Hyperledger Fabric Development with Composer"



Hyperledger Concepts

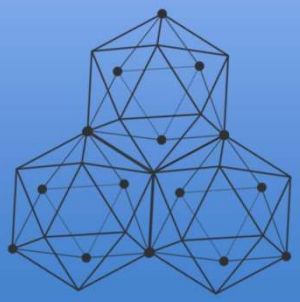
raj@acloudfan.com

@acloudfan

http://ACloudFan.com

Learning Objectives:

- Assets
- Chaincode
- Ledger



Assets

- JSON or Binary

```
{
  vinNumber: …
  owner: ….
}
```

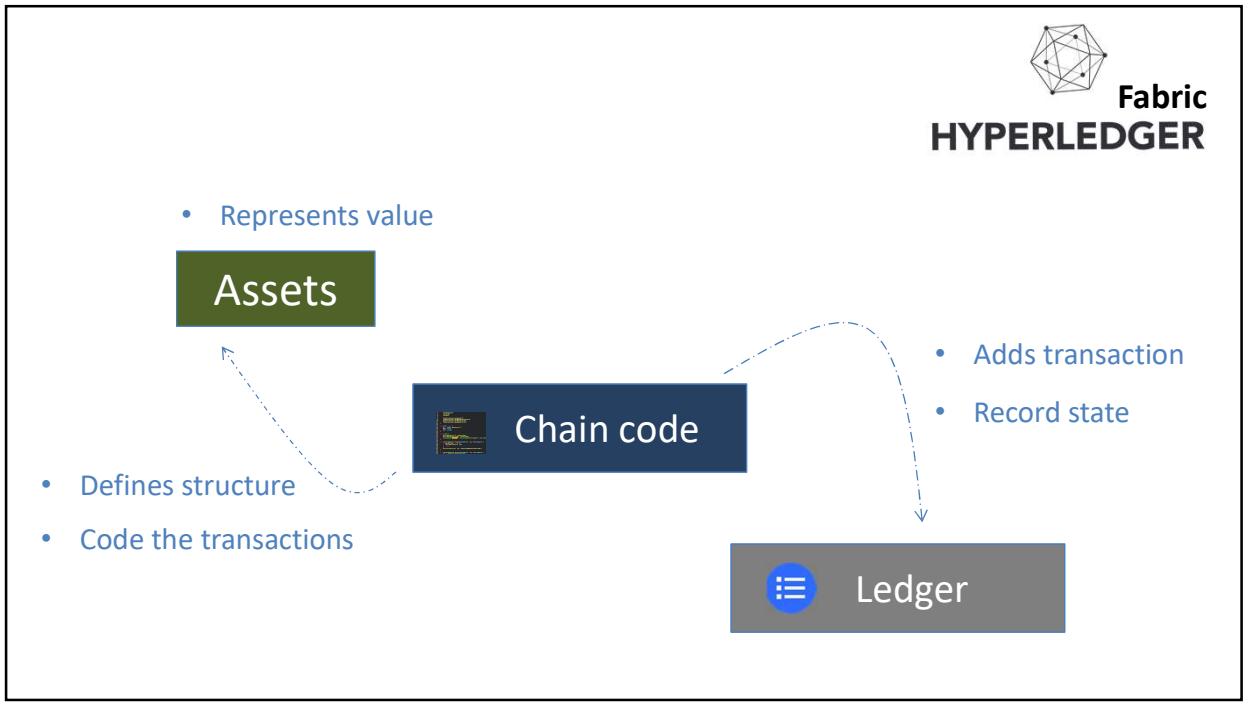
Chain code

- Defines the asset's structure
- Transaction | Business logic

Changes State of Asset
Function sellTheCar(newOwner)

Ledger

- Tracks all of the asset Transactions
  - Records State changes of the asset
- Ledger is distributed (DLT)
  - All participants have a replica of the ledger



Hyperledger Concepts

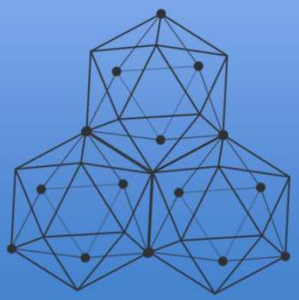
raj@acloudfan.com

@acloudfan

http://ACloudFan.com

Learning Objectives:

- Permissioned network
  - Identities
  - Membership Service Provider



Businesses interact with known entities

ACME Corp.

ABC Corp.

John Doe

Regulatory

Anonymous Access

Permissioned network

All identities are known

Membership Service Provider

Roles

Access Restrictions

Member = Legally separate entities

Manufacturer

A

B

C

Dealer

Repair Shop

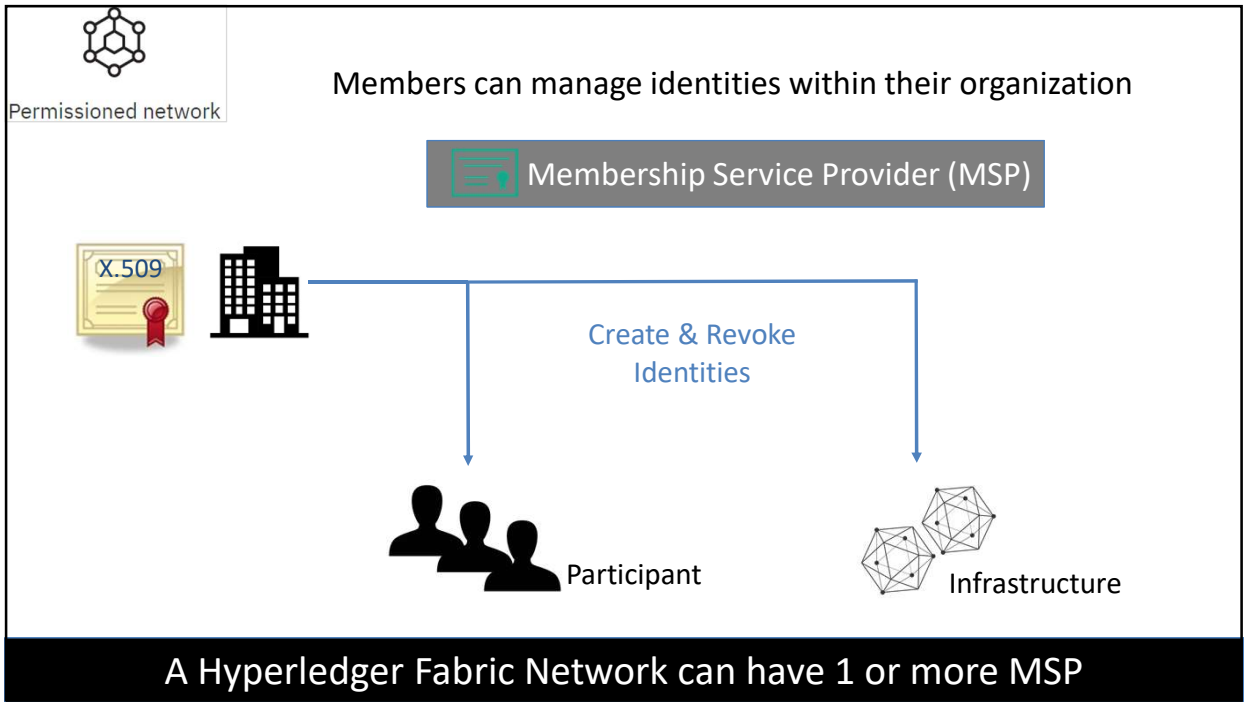
Permissioned network

Identity managed by way of certificates

Infrastructure

Member

Certificates issued & revoked by Certification Authority (CA)



A Hyperledger Fabric Network can have 1 or more MSP

Membership Service Provider

- Abstract component that manages identities
- Provides the credentials to various entities

- Pluggable
  - Network may have 1 or more MSP

Hyperledger Concepts

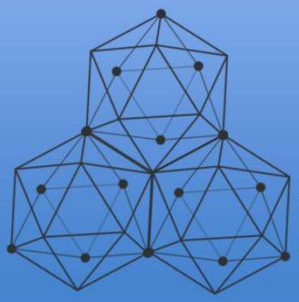
raj@acloudfan.com

@acloudfan

http://ACloudFan.com

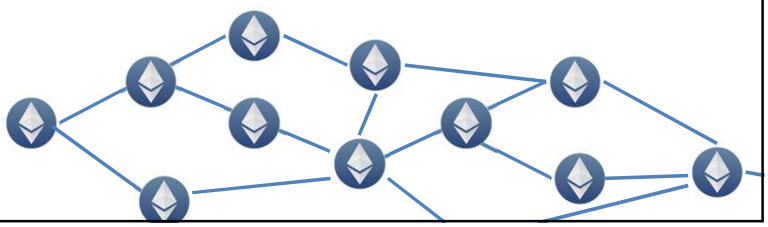
Learning Objectives:

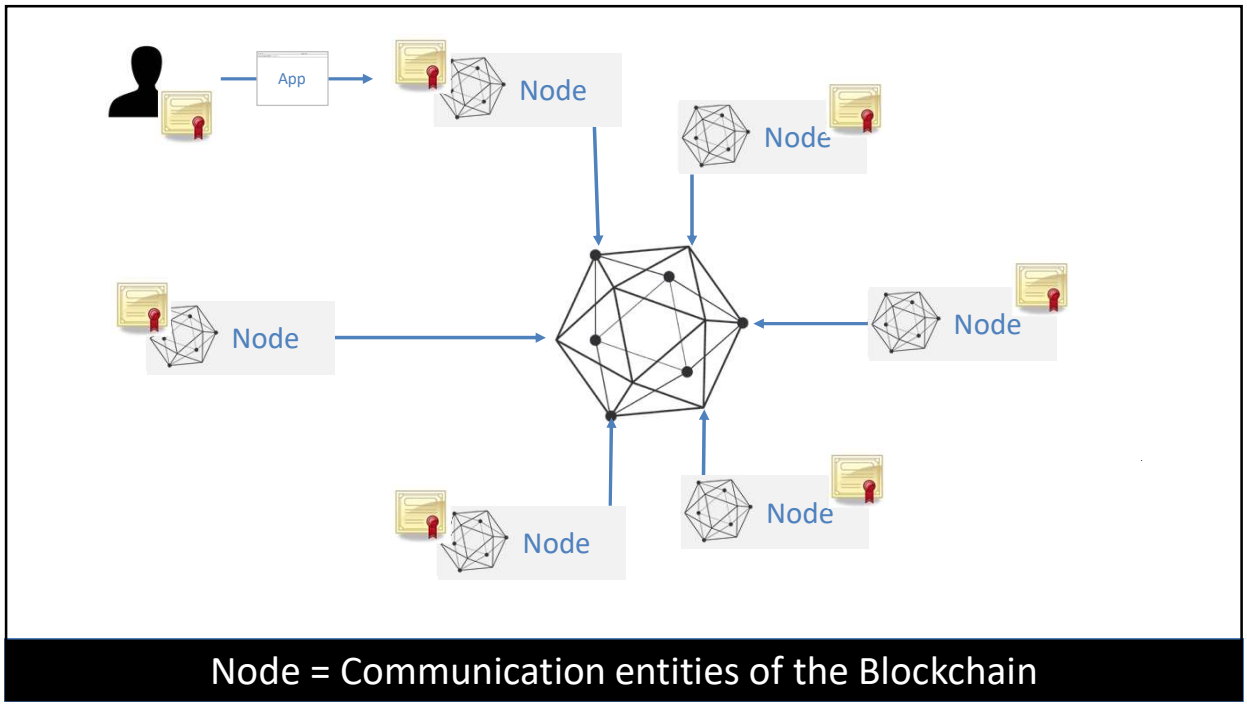
- Nodes
- Channels



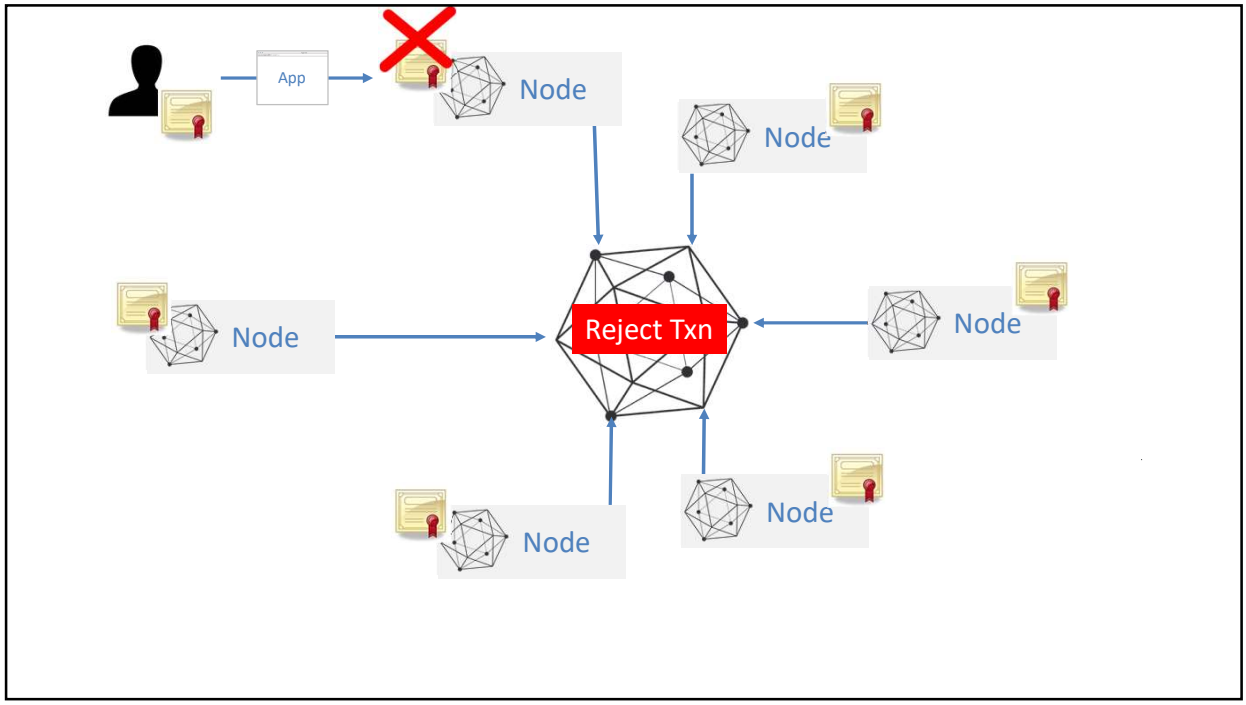
- All BC technologies have the concept of nodes
  - Nodes connect to other nodes to form the BC network

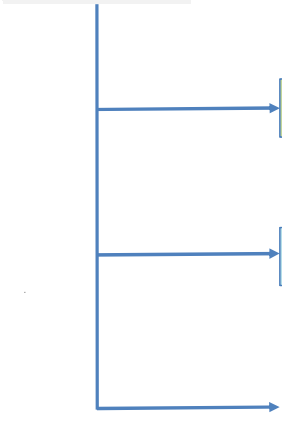
ETHEREUM bitcoin All Nodes are Equal





Node = Communication entities of the Blockchain



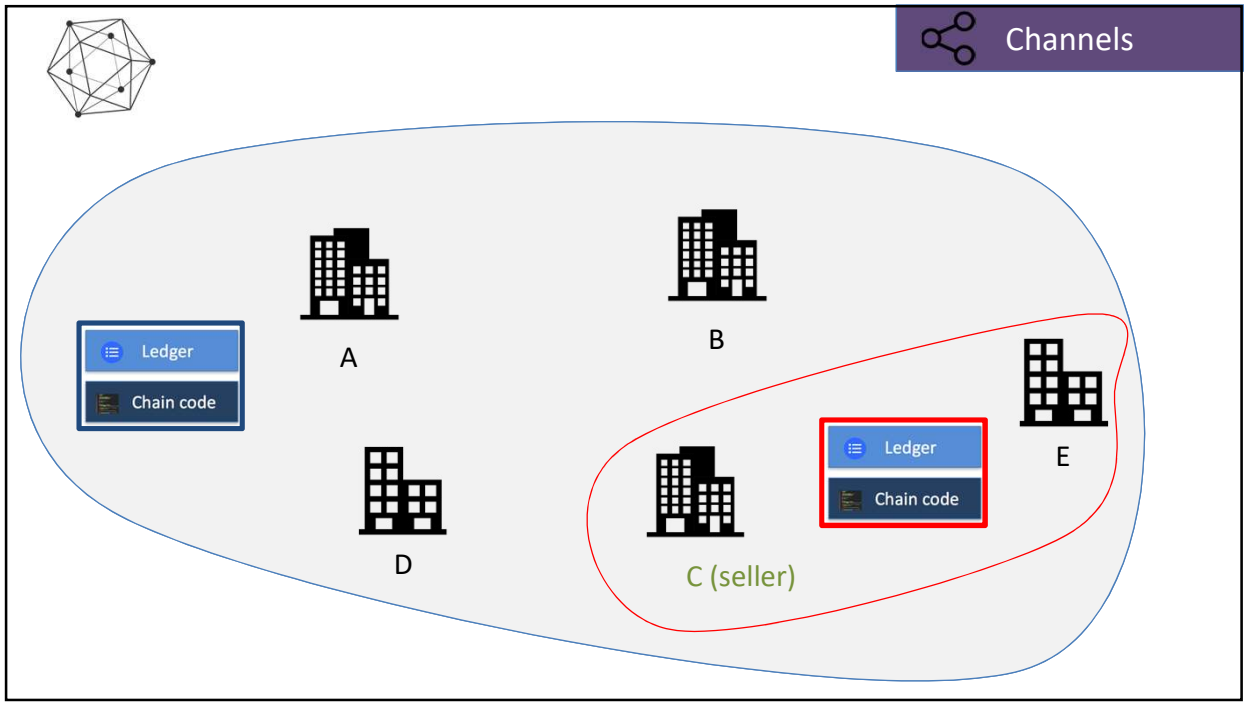


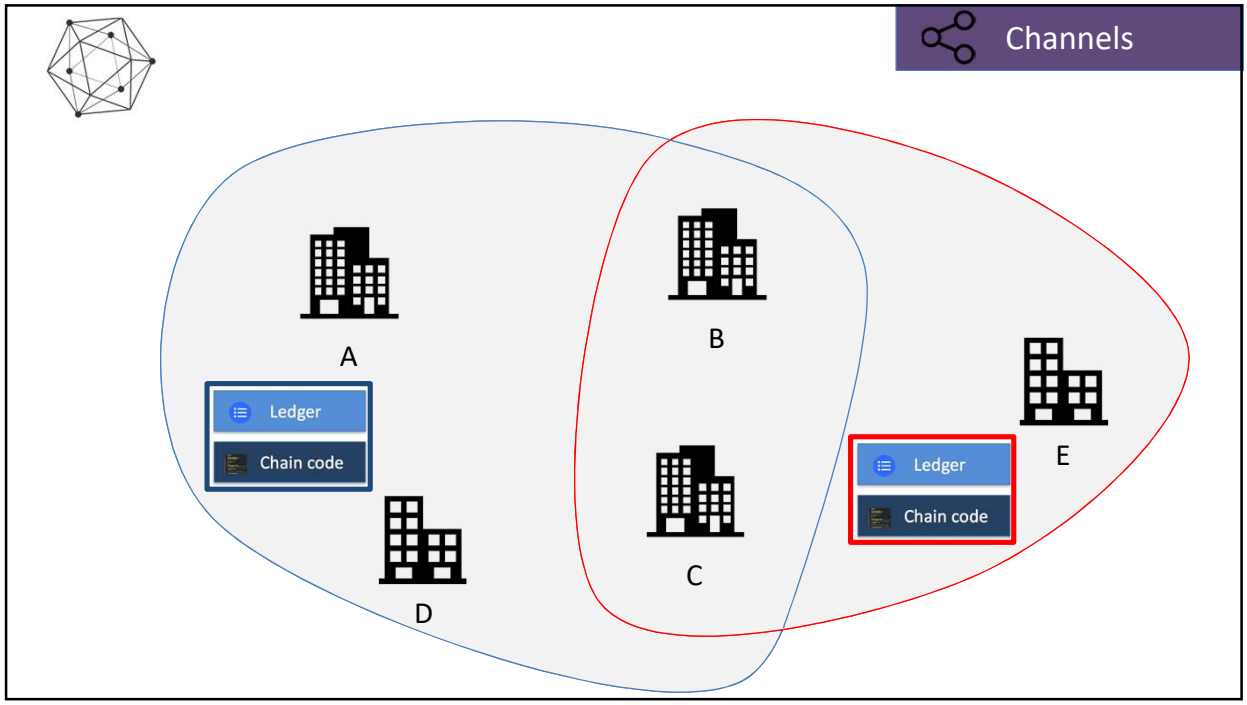
- Responsible for distribution of transactions
- Keeps the Ledger data in synch across network
- Initiates the transactions

In Hyperledger Fabric = All nodes are **NOT** equal

- Members can participate on multiple HL BC Networks
  - Transactions in each network is isolated

- Peers connect to the channel
- Independent Ledger in each channel

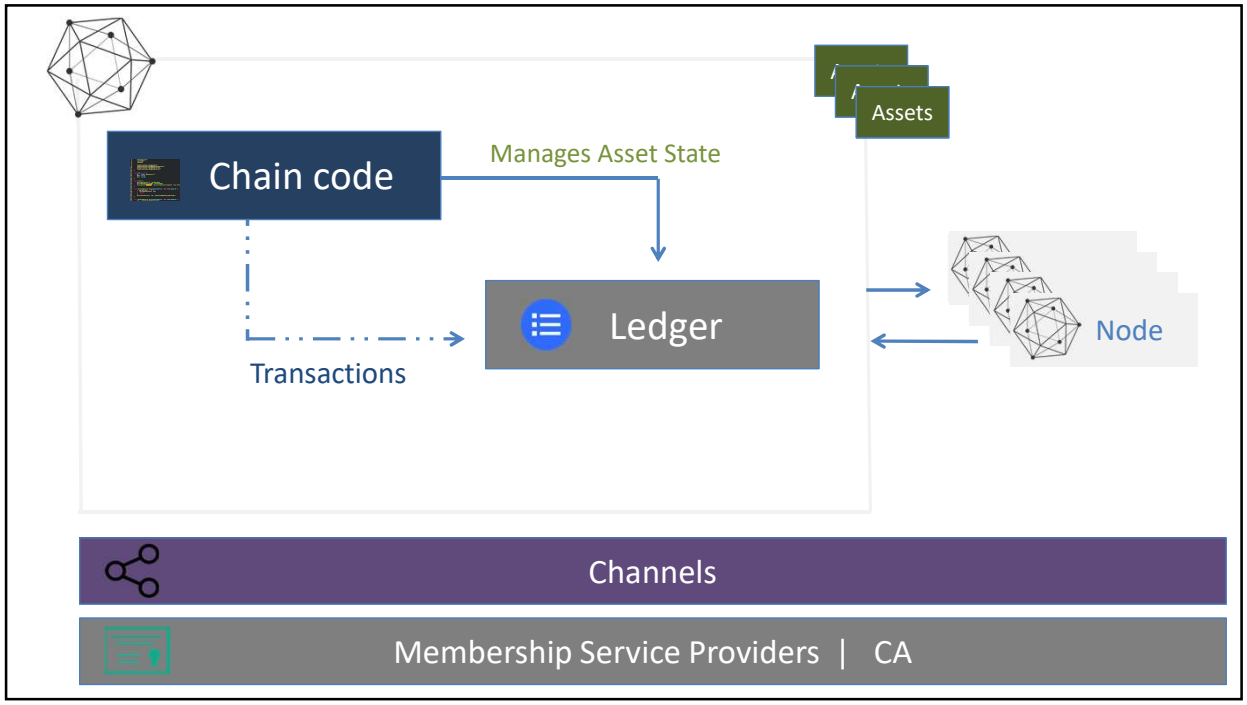

Channels
Ledger
Chain code
A
B
D
Ledger
Chain code
E
C (seller)


Channels
A
Ledger
Chain code
D
B
C
Ledger
Chain code
E

privacy

1. By way of Private channel
   - Isolates the Ledger | Transactions
2. Intermediate solution
   - Common channel
   - Chaincode installed on peers that need visibility

+ Use Data Encryption



Consensus

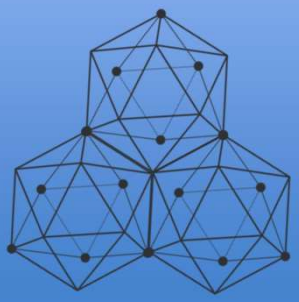
raj@acloudfan.com

@acloudfan

http://ACloudFan.com

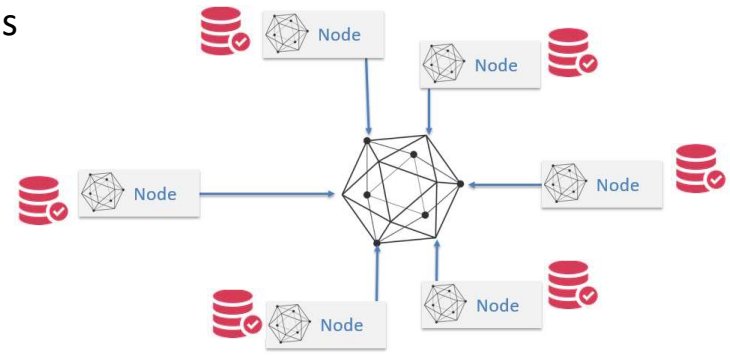
Learning Objectives:

- High Level Overview
  - Chaincode
  - State
  - Ledger



Consensus

- Ensures consistency of ledger data across nodes
  - All approvers agree to the transaction
- Order of transactions



Consensus

- Implemented as node referred to as Ordering Service
- Pluggable
  - Members in the network decide
  - New models easy to implement

- No concept of mining
- No need to incentivize
- No crypto token

Ledger Implementation

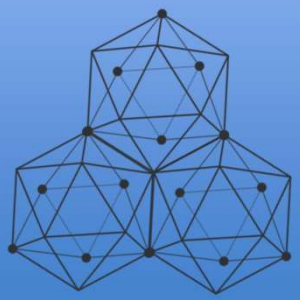
Learning Objectives:

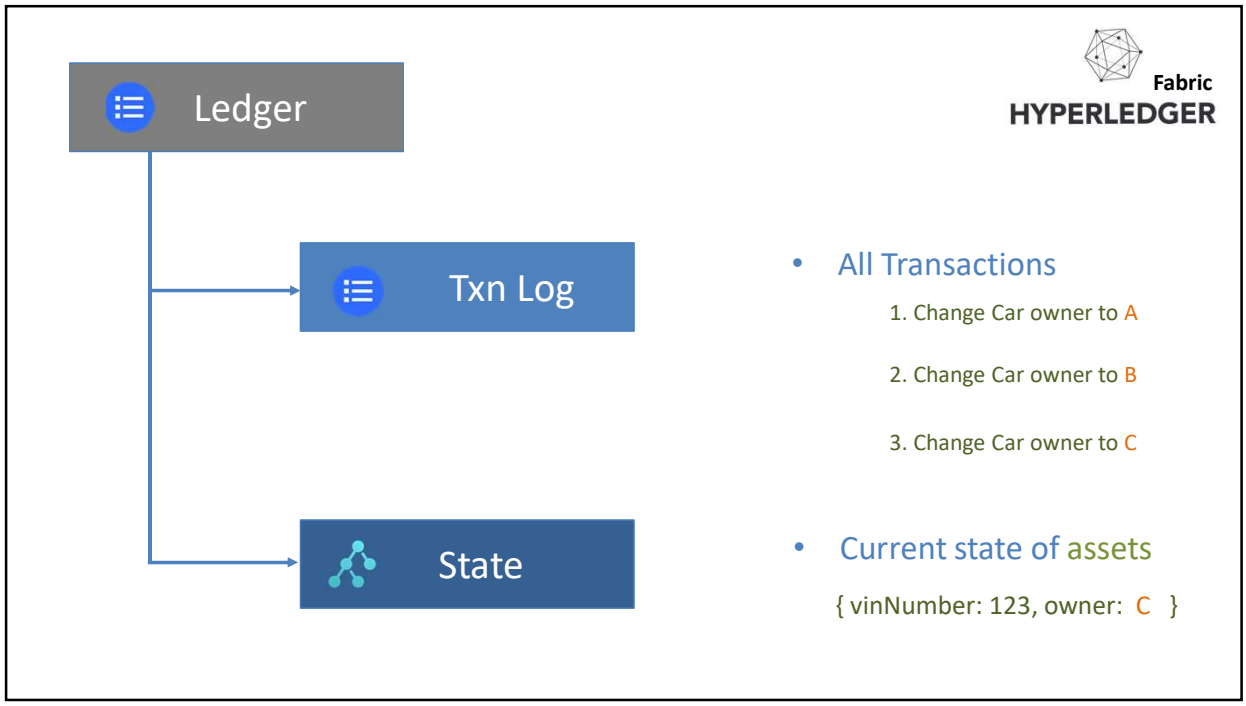
- Transaction Log
- State database

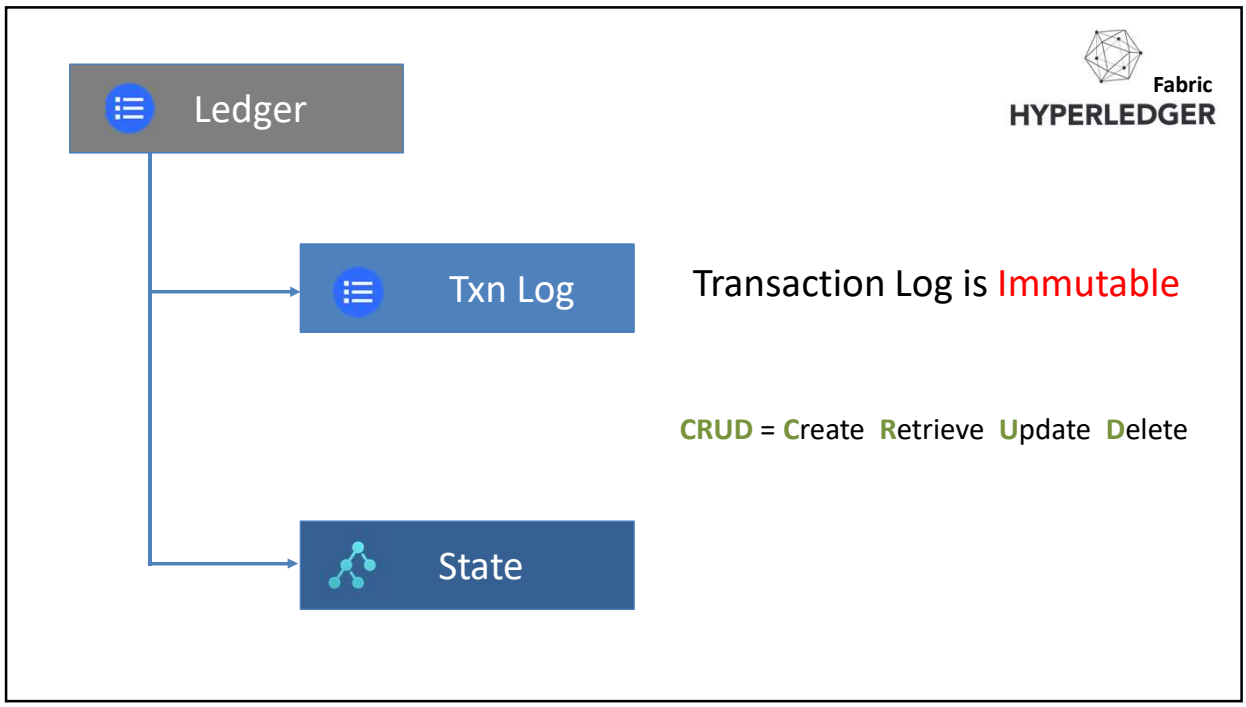
raj@acloudfan.com

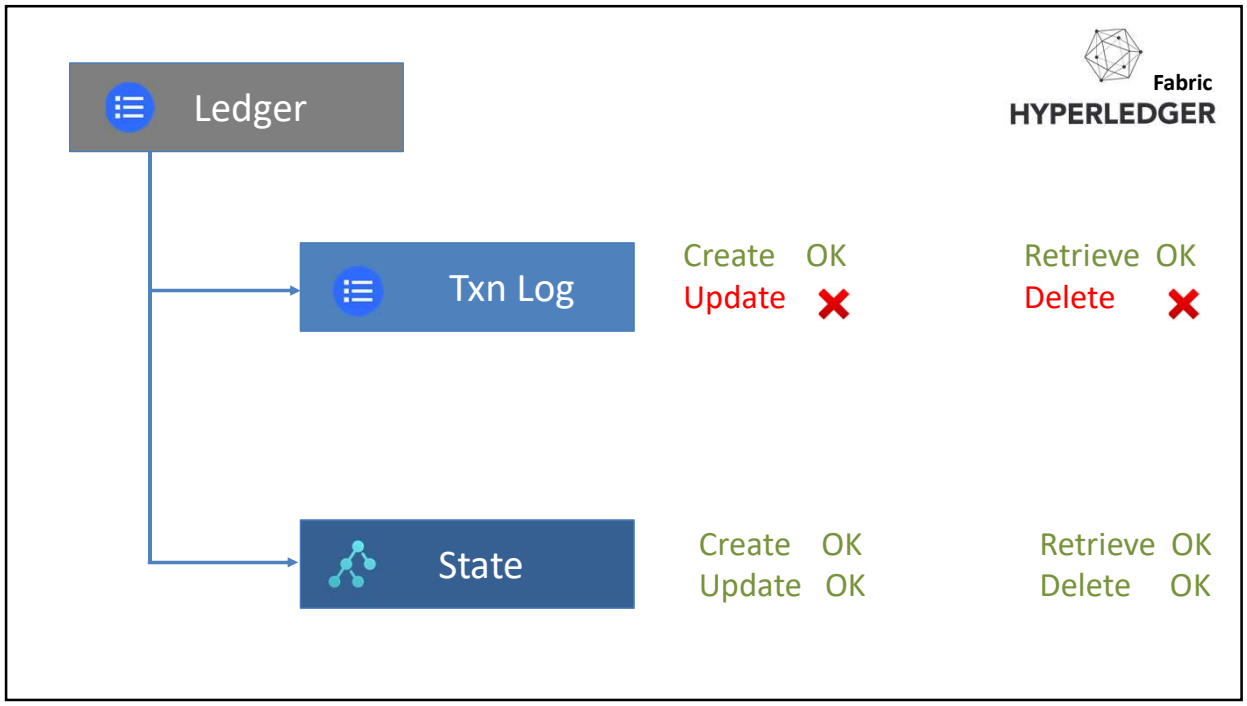
@acloudfan

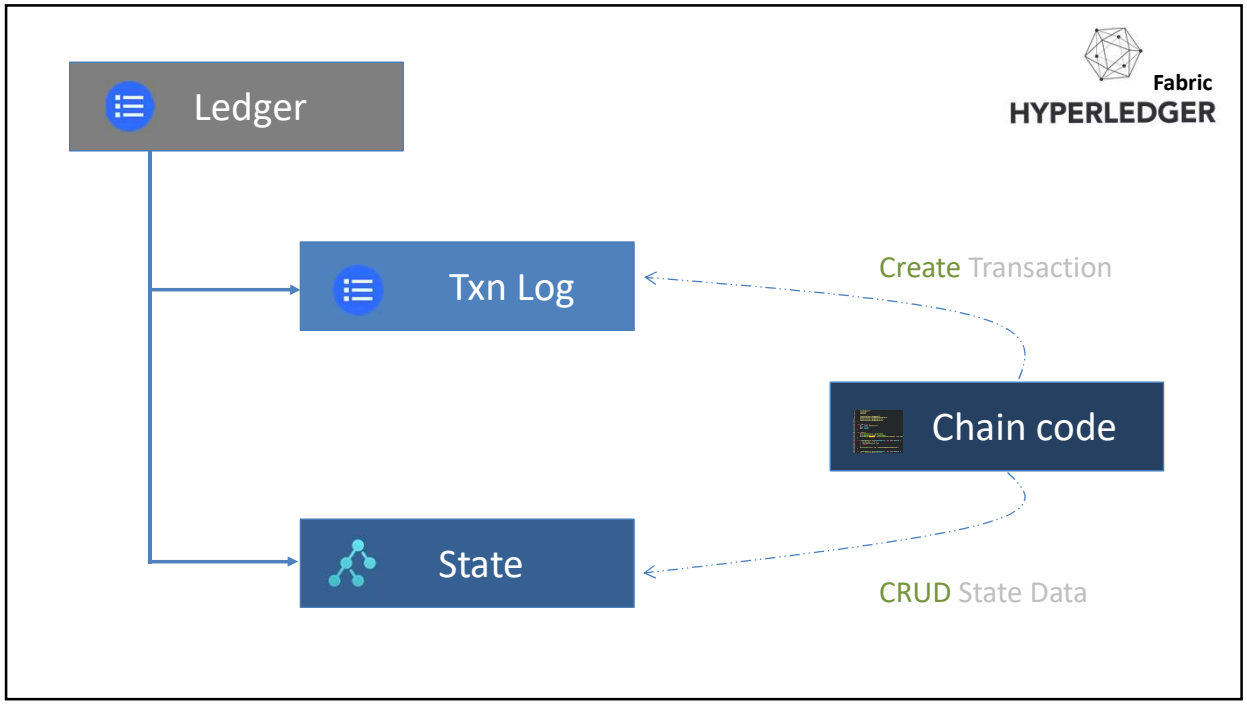
http://ACloudFan.com

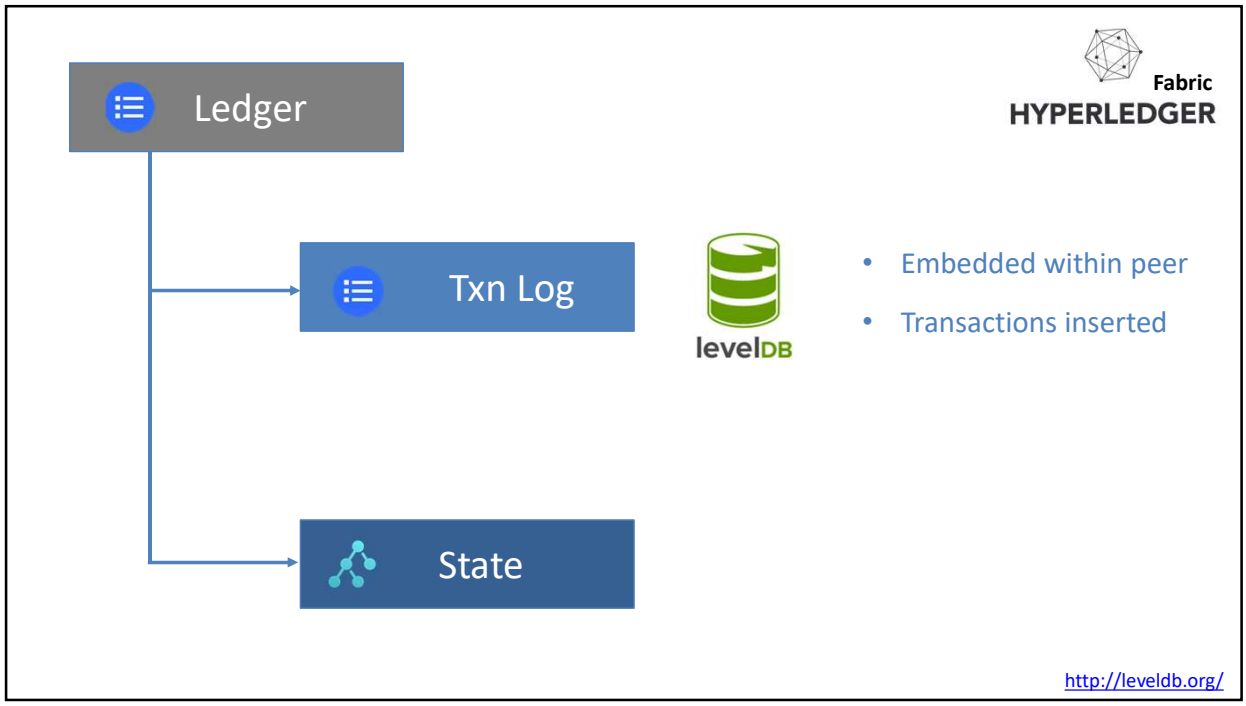


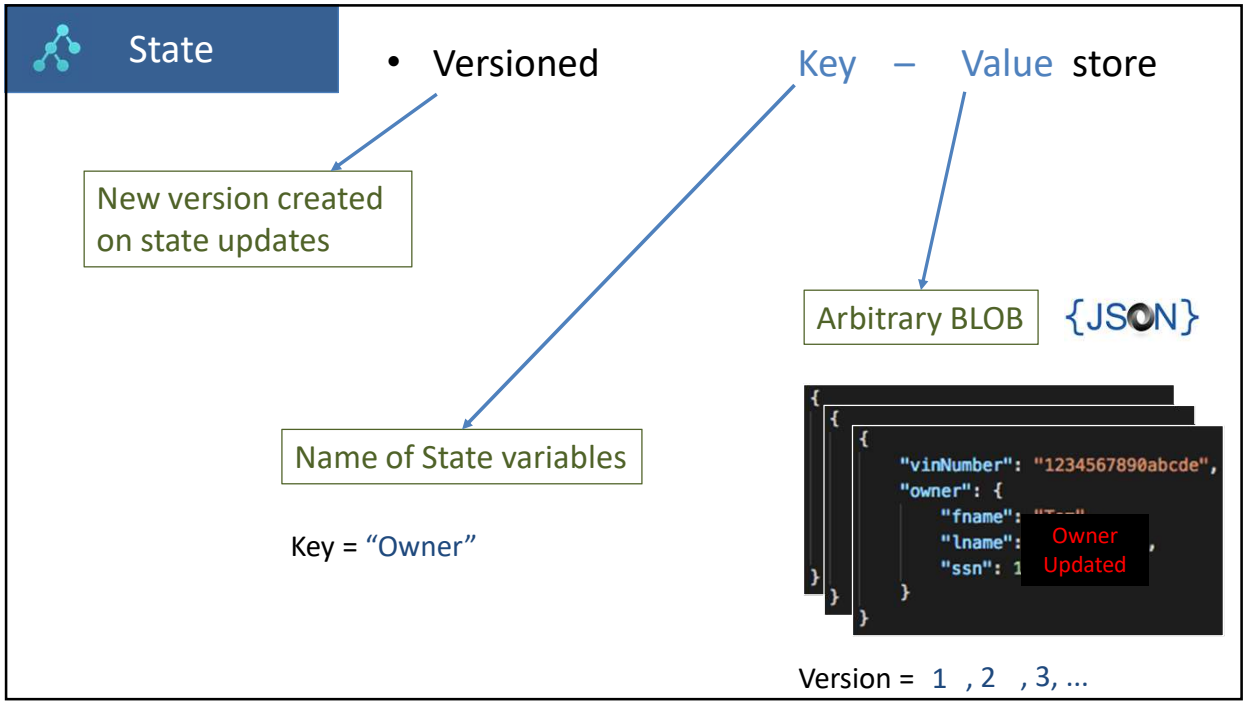

Ledger
Txn Log
State
Fabric
HYPERLEDGER
All Transactions
1. Change Car owner to A
2. Change Car owner to B
3. Change Car owner to C
Current state of assets
{ vinNumber: 123, owner: C }


Ledger
Txn Log
State
Fabric
HYPERLEDGER
Transaction Log is Immutable
CRUD = Create Retrieve Update Delete


Ledger
Fabric
HYPERLEDGER
Txn Log
Create OK
Update
Retrieve OK
Delete
State
Create OK
Update OK
Retrieve OK
Delete OK


Ledger
Fabric
HYPERLEDGER
Txn Log
Create Transaction
Chain code
State
CRUD State Data

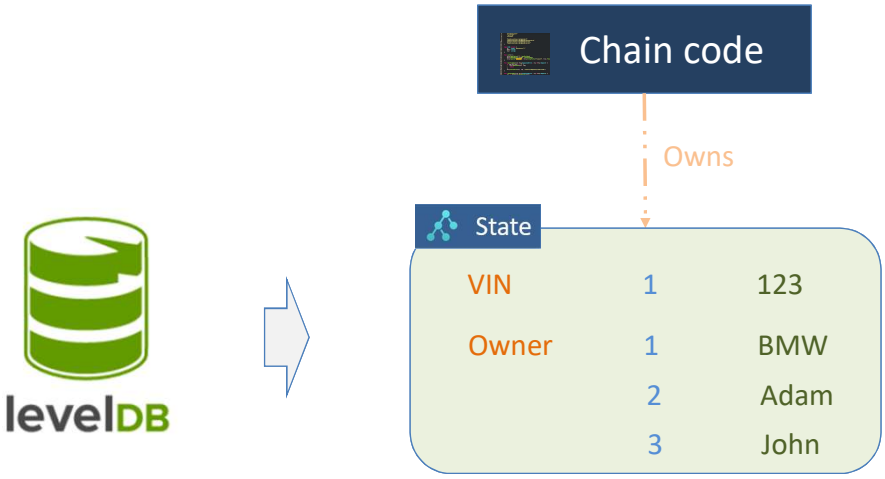

Ledger
Txn Log
State
Fabric
HYPERLEDGER
levelDB
• Embedded within peer
• Transactions inserted
http://leveldb.org/


State
• Versioned
Key – Value store
New version created on state updates
Arbitrary BLOB
{JSON}
Name of State variables
Key = "Owner"
"vinNumber": "1234567890abcde",
"owner": {
"fname":
"lname":
"ssn":
Owner Updated
Version = 1 , 2 , 3, ...

State

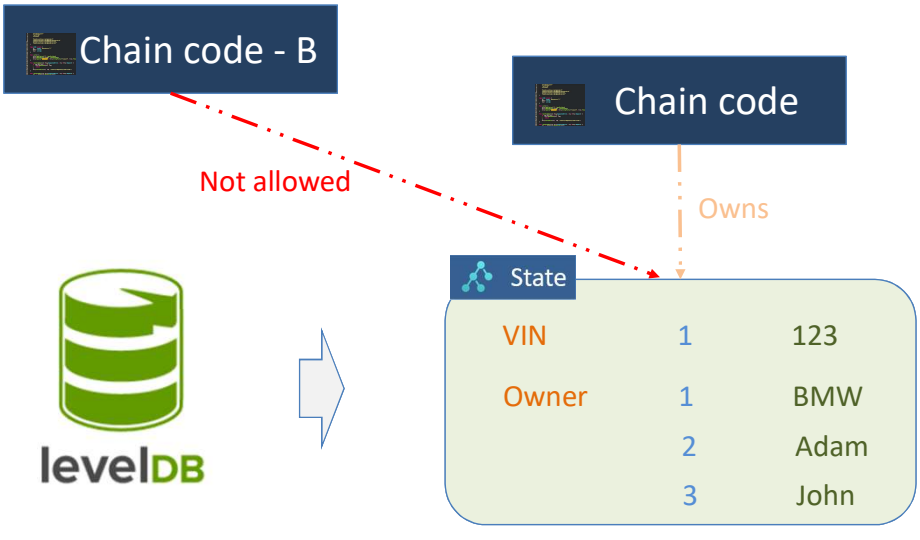
State(Key) = (Version, Value)

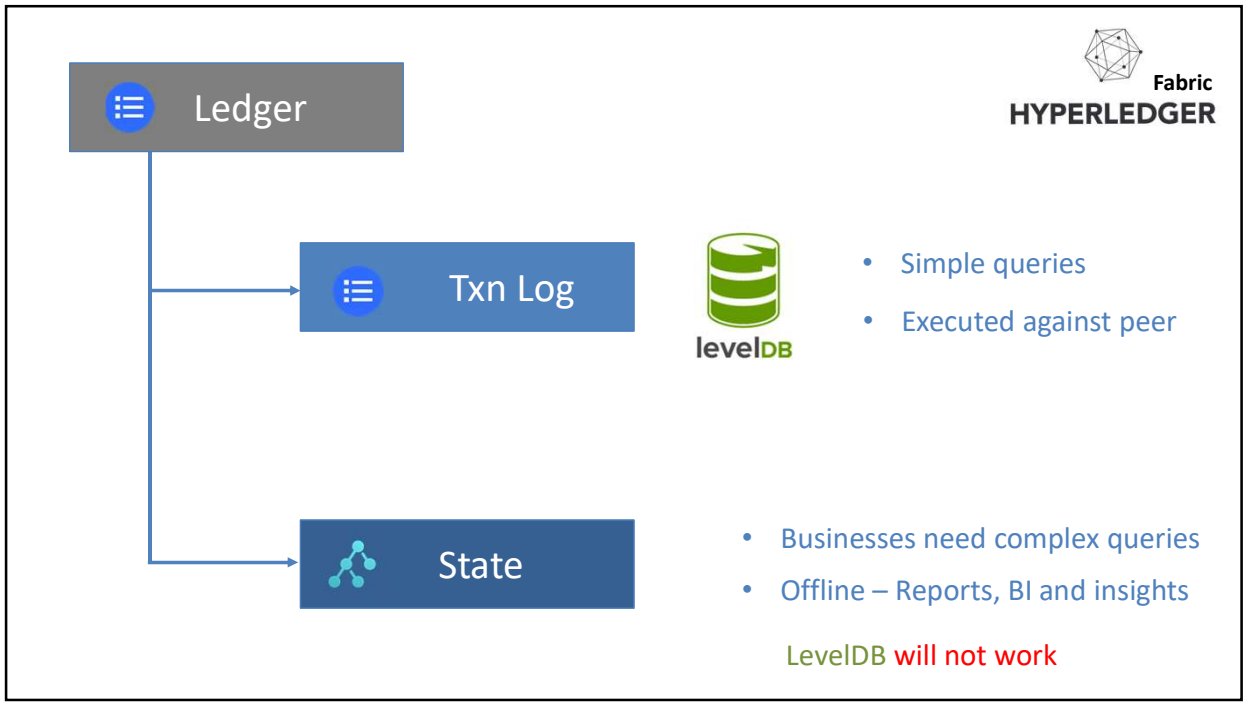
- By default State Data is managed in LevelDB

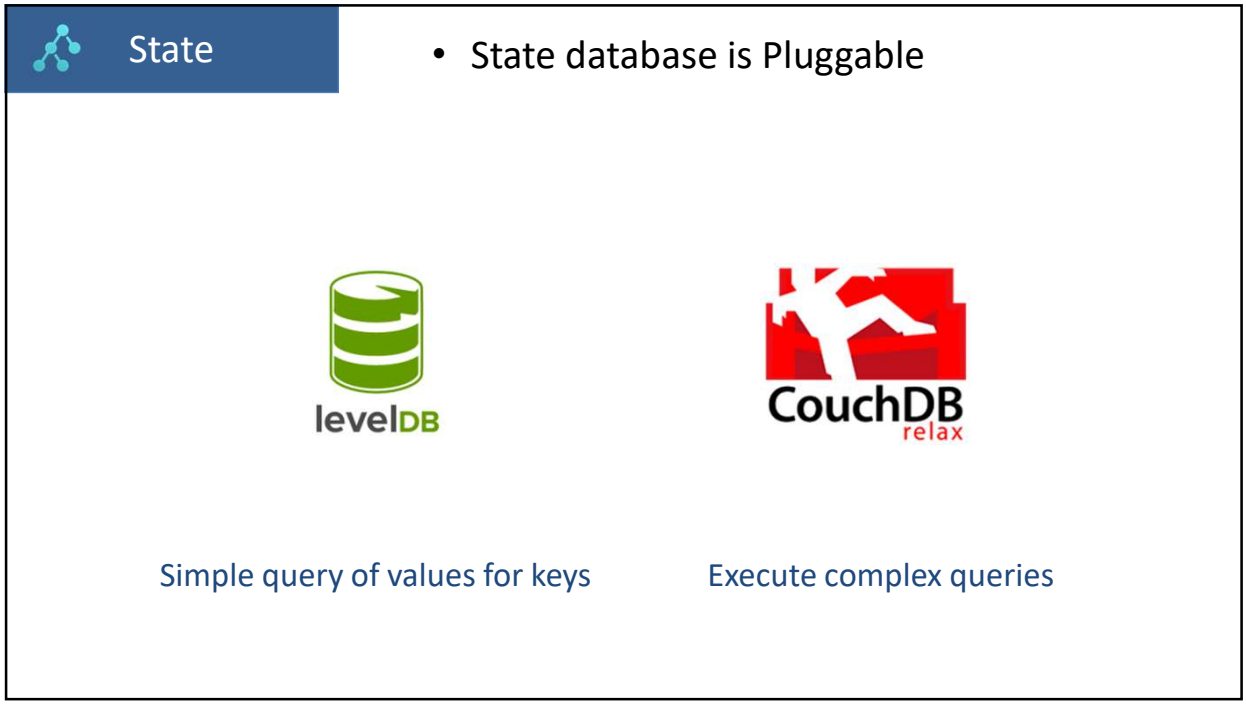


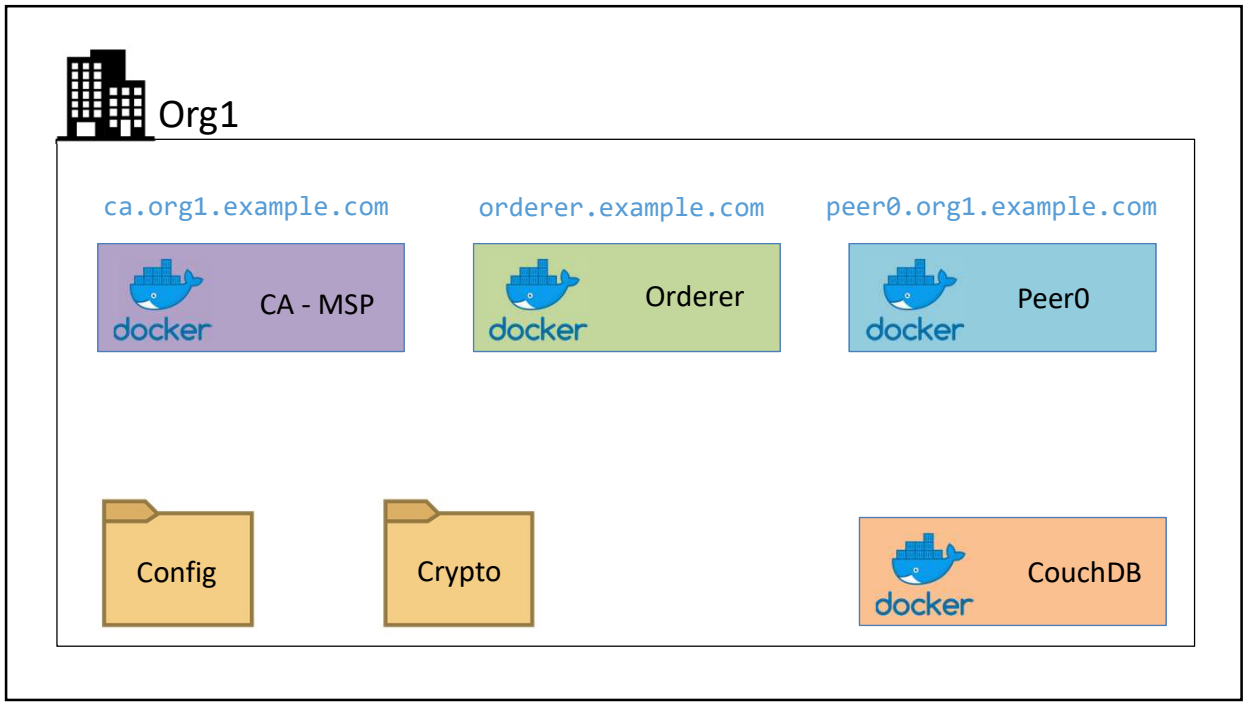
State

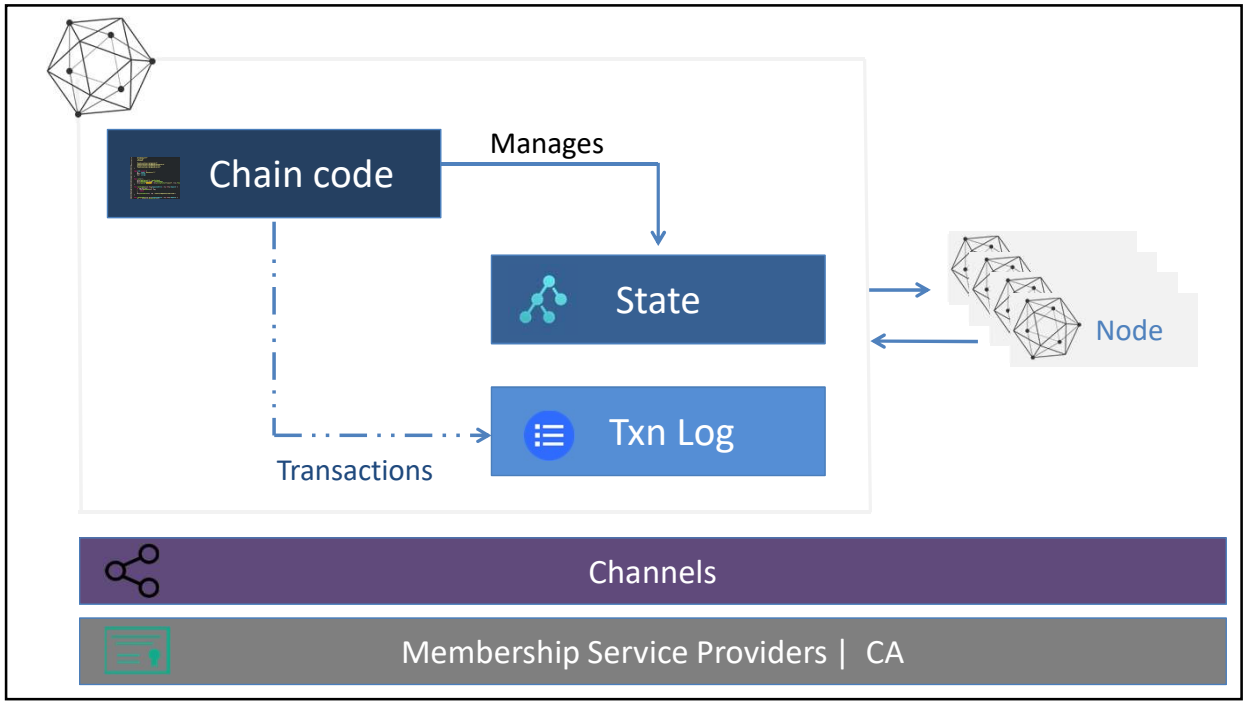
Access restricted to owner chaincode




Ledger
Txn Log
State
Fabric
HYPERLEDGER
levelDB
• Simple queries
• Executed against peer
• Businesses need complex queries
• Offline – Reports, BI and insights
LevelDB will not work


State
• State database is Pluggable
levelDB
CouchDB
relax
Simple query of values for keys
Execute complex queries


Org1
ca.org1.example.com
orderer.example.com
peer0.org1.example.com
docker
CA - MSP
docker
Orderer
docker
Peer0
Config
Crypto
docker
CouchDB


Chain code
Manages
State
Node
Txn Log
Transactions
Channels
Membership Service Providers | CA

Chaincode

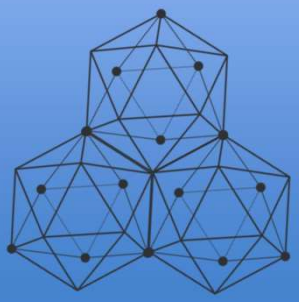
raj@acloudfan.com

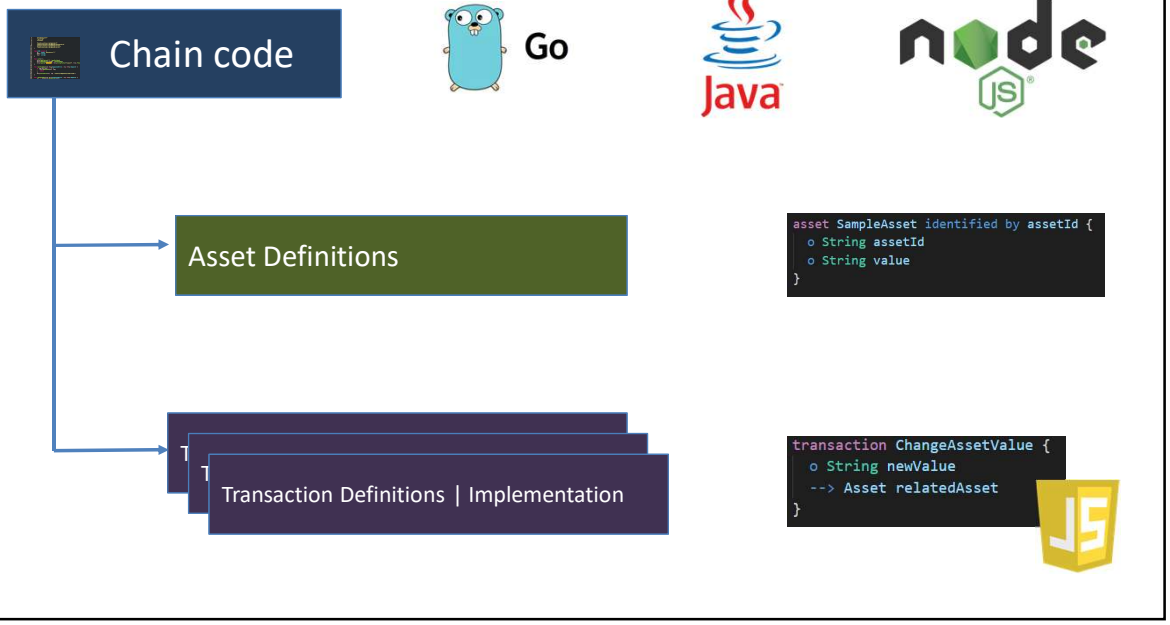
@acloudfan

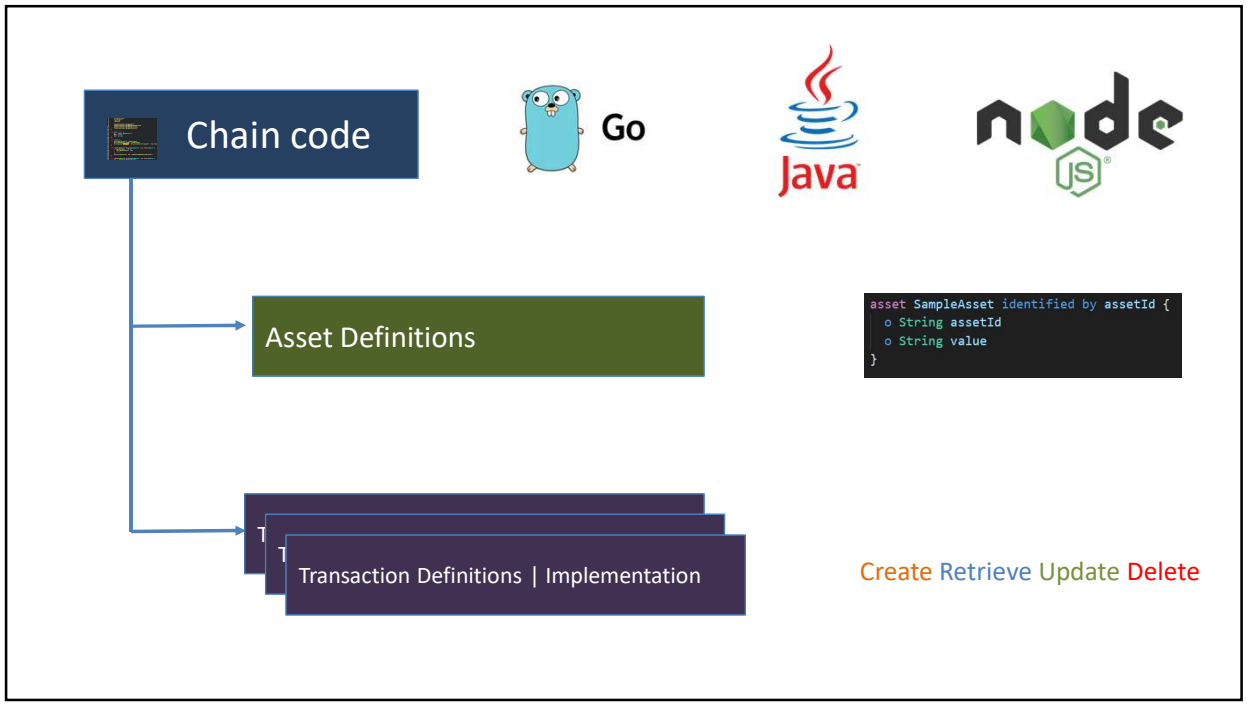
http://ACloudFan.com

Learning Objectives:

- Structure
- Development workflow
- Execution runtime







What is Hyperledger?

Discount Coupon Links to UDEMY courses:


https://www.udemy.com/hyperledger/?couponCode=DKHLF1099


https://www.udemy.com/ethereum-dapp/?couponCode=DKETH1099


https://www.udemy.com/rest-api/?couponCode=DKRST1099

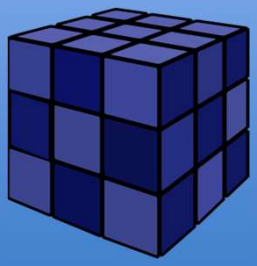
mentoring, seeking Blockchain part time work, project guidance, advice … …
http://www.bcmentors.com

This deck is part of a online course on "Hyperledger Fabric Development with Composer"

raj@acloudfan.com

@acloudfan

http://ACloudFan.com

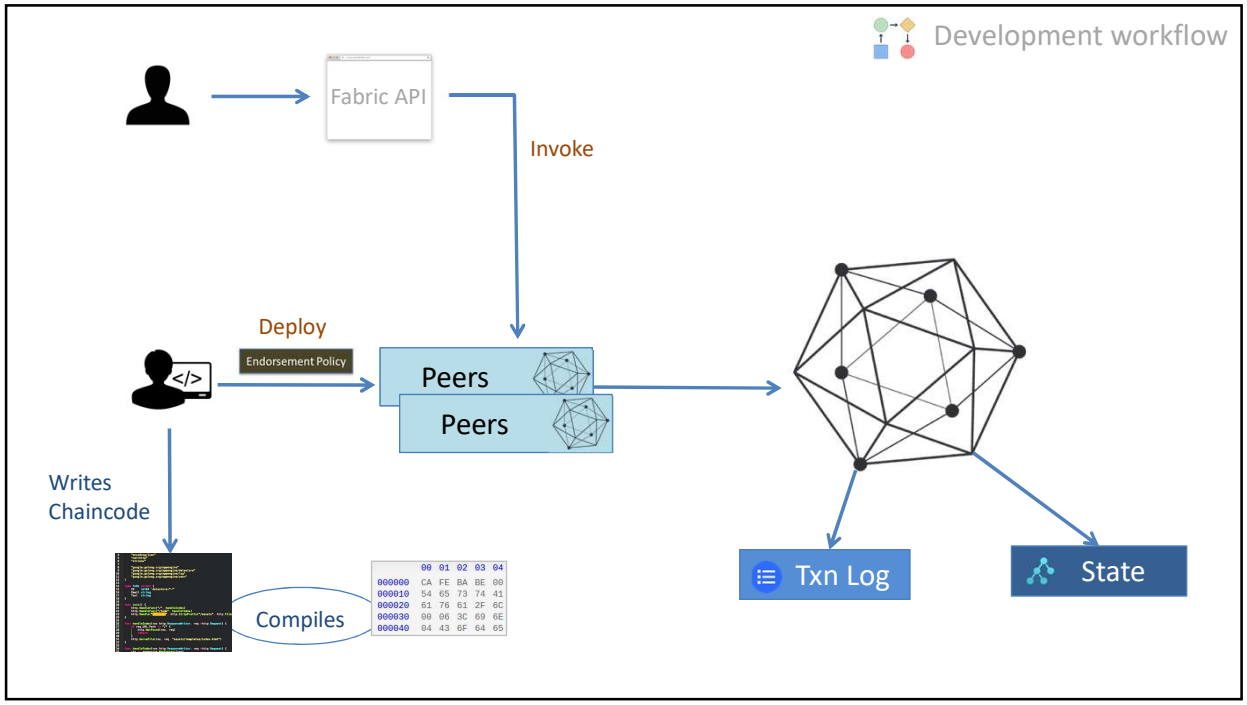


Chain code

- May specify an Endorsement Policy

Peers

- Instruction on how to validate the transaction



Peer Nodes

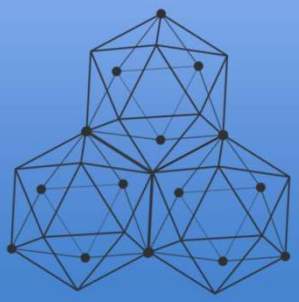
raj@acloudfan.com

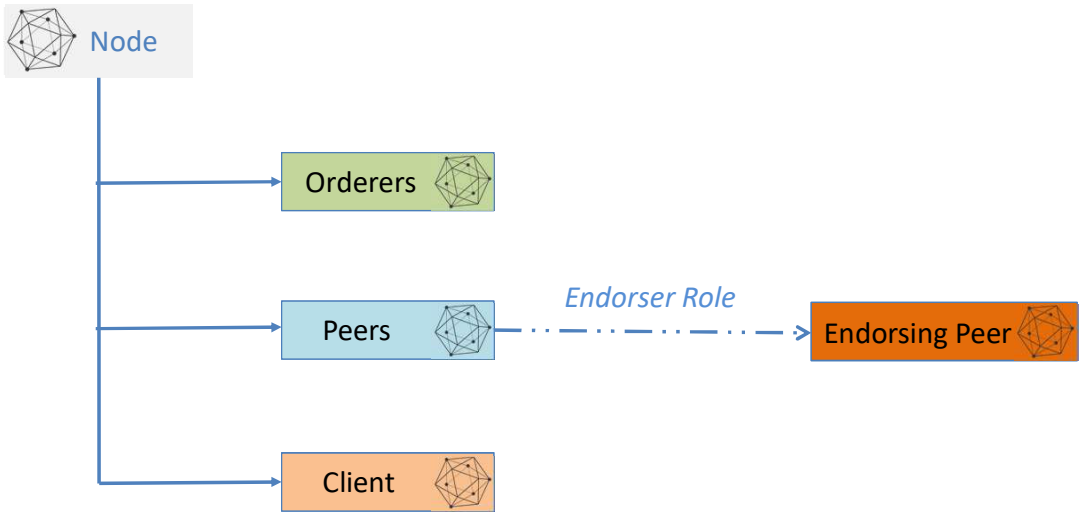
@acloudfan

http://ACloudFan.com

Learning Objectives:

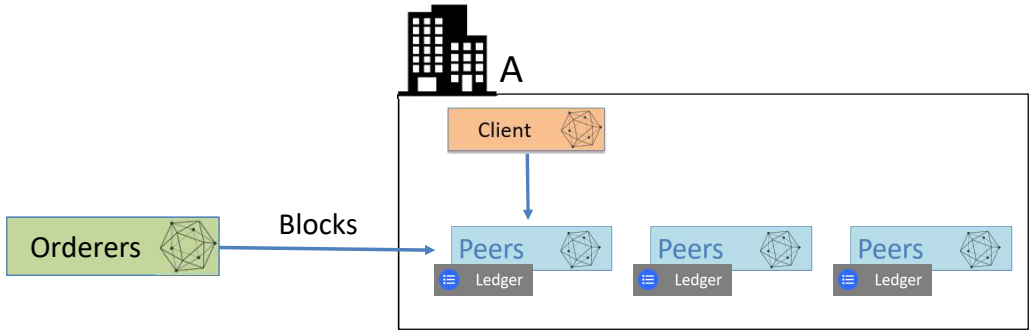
- Peers
  - Anchor peers
  - Endorsing peers





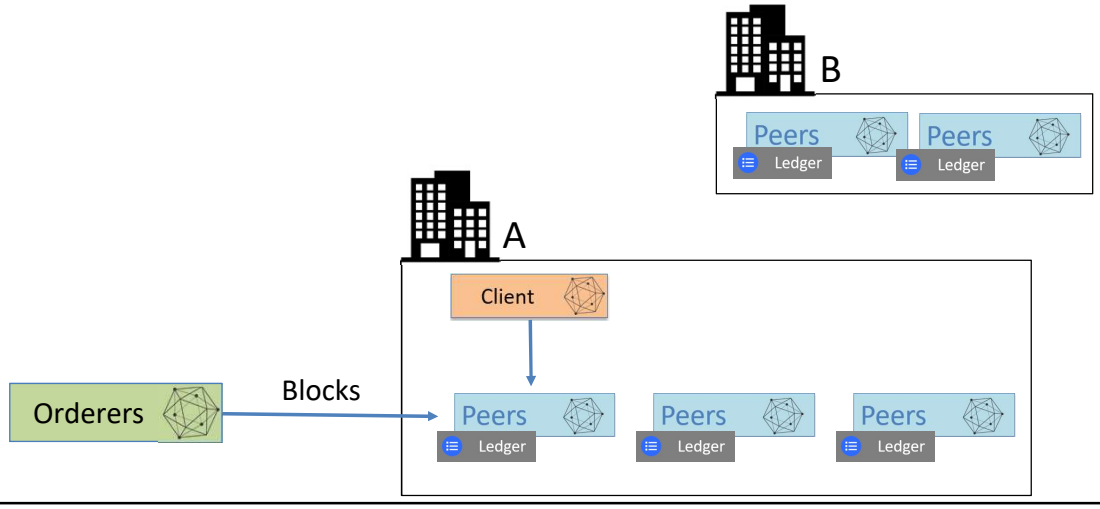
Peers

- Receives requests from client
- Local Ledger Synchronization



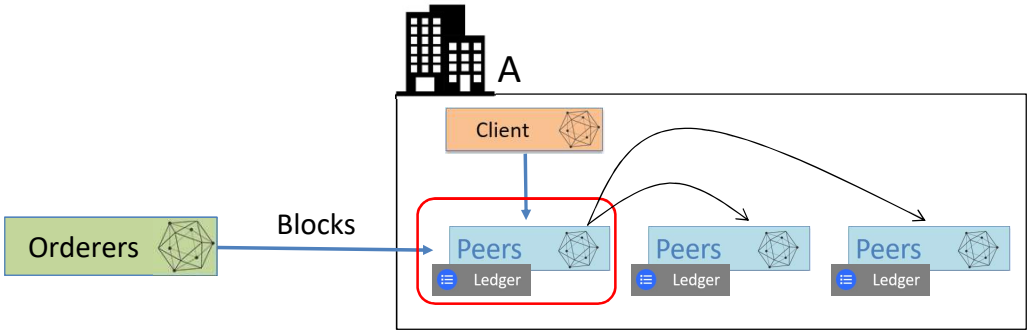
Peers

- Highly Scalable Architecture
  - No centralized effort needed



Peers

- **Anchor Peers** receive the blocks
  - Anchor Peers update other peers in the org
  - Setup at the channel level
  - Discoverable



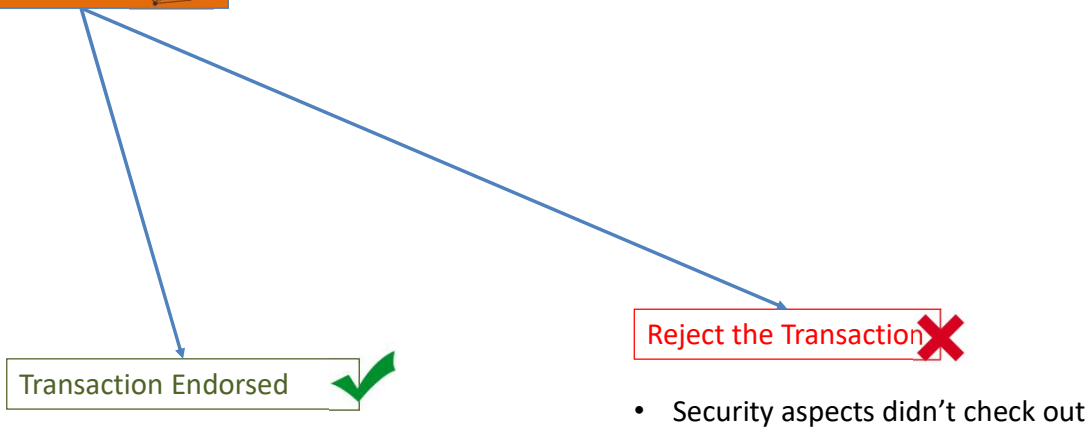
Endorser

- Peer marked as the endorser a.k.a. endorsing peer
  - Validates the transaction e.g., Certificate checks
  - Simulates the chaincode
    - **Executes** the code
    - But does **NOT** save the state to the Ledger

Endorser

- Primary objective = Protect the network
  - Intentional attack on the network
  - Misbehaving or misconfigured nodes on the network
- Improve scalability as only endorsers need to execute the code
  - NO need for all nodes to execute the chaincode

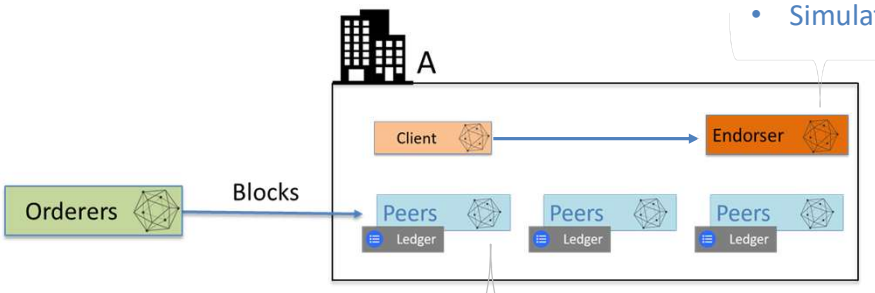
Endorser



- Security aspects didn't check out
- Execution failure

Summary

- Protects the network
- Rejects | Accepts
- Simulates the transaction



Anchor Peer

- Receives blocks
- Provide blocks to other peers in the organization