NATIONAL INSTITUTE OF TRANSPORT

Blockchain Technology Short Training

Introduction to Hyperledger Fabric

Facilitator: Dr. Cleverence Kombe (PhD)

Introduction to Hyperledger | Overview

- Nodes in Hyperledger Fabric Network
- Endorsement policy
- Hyperledger Fabric runtime architecture
  - Transaction proposal (Client SDK)
  - Transaction endorsement
  - Transaction submitted to the ordering service
  - Transaction validation
- Chaincode lifecycle
- System chaincode

Hyperledger Fabric Network | Nodes

- In a Hyperledger Fabric blockchain network, nodes play an important role in validating transactions and maintaining the integrity of the network.
- There are several types of nodes in a Hyperledger Fabric network, including:
  - ***Peer nodes***
  - ***Orderer nodes***
  - ***Client nodes***
  - ***Certificate Authority nodes***
  - ***Admin nodes***
  - ***Anchor nodes***

Hyperledger Fabric Network | Peer Nodes

- Peers are nodes that maintain a copy of the distributed ledger and participate in the consensus protocol to validate transactions and reach agreement on the state of the ledger.
- Peers can be categorized into two types: **Endorsing peers** and **Committing peers.**
  - **Endorsing Peers:** When a client initiates a transaction, the transaction proposal is sent to the endorsing peers. The endorsing peers simulate the transaction and check if it meets the endorsement policy set for the chaincode. If it does, they endorse the transaction by signing it with their private key and return it to the client.

Hyperledger Fabric Network | Peer Nodes

- Peers are nodes that maintain a copy of the distributed ledger and participate in the consensus protocol to validate transactions and reach agreement on the state of the ledger.
- Peers can be categorized into two types: **Endorsing peers** and **Committing peers.**
  - **Committing Peers:** The committing peers receive the endorsed transactions from the endorsing peers. They validate the transactions, ensure that they meet the endorsement policy, and check if they are valid according to the rules set for the chaincode. If everything checks out, they commit the transaction to the ledger and broadcast the new state to all the other peer nodes.

Hyperledger Fabric Network | Orderer Nodes

- Orderer nodes are responsible for managing the ordering of transactions in the network.
- They receive endorsed transactions from the committing peers, package them into blocks, and broadcast the blocks to all the peer nodes.
- They use different consensus algorithms, such as Kafka or Raft.
- They can be organized into a cluster, which provides fault-tolerance and high availability for the ordering service.
- They are responsible for creating and distributing the Genesis block that initializes the network.

Hyperledger Fabric Network | Certificate Authority (CA) Nodes

- Certificate Authority (CA) nodes manage the digital certificates used to authenticate users and nodes in the network.
- They issue certificates to nodes and users when they join the network, and revoke certificates when necessary.
- They can be configured to use different certificate signing algorithms and expiration policies.
- They use the Fabric CA server, which is an implementation of the Certificate Management Service (CMS) standard.
- The CA nodes also manage the certificate revocation lists (CRLs), which keep track of all the revoked certificates in the network.

Hyperledger Fabric Network | Client Nodes

- They interact with the network by submitting transaction proposals to orderer nodes and receiving responses from peer nodes.
- They use software development kits (SDKs) to communicate with the network.
- They do not maintain a copy of the ledger, but can query the ledger for information.
- They can be either user clients, which represent end users of the network, or application clients, which represent software applications that interact with the network.

Hyperledger Fabric Network | Admin Nodes

- They can manage the network and its resources, including creating channels, adding or removing nodes, and configuring network settings.
- They can use administrative tools, such as the Hyperledger Fabric command line interface (CLI) or the Hyperledger Fabric SDK for Node.js, to interact with the network.
- They can be authorized to perform specific tasks using Fabric policies and access control lists (ACLs)..
- They require special permissions and digital certificates to access sensitive network resources.

Hyperledger Fabric Network | Anchor Nodes

- They can be used in multi-organization networks to establish a communication channel between different organizations.
- They serve as a gateway for communication between peer nodes in different organizations.
- Anchor nodes in each organization maintain a record of the organization's peers that are participating in the channel.
- Updates to the anchor node's list of peers are broadcast to other anchor nodes in the network.

Hyperledger Fabric Network | Endorsement policy

- Endorsement policies are an important feature of the Hyperledger Fabric blockchain network. They define the criteria that must be met for a transaction to be considered valid and added to the blockchain.
- In Hyperledger Fabric, transactions are proposed by clients and sent to the endorsing peers for validation. The endorsement policy determines which peers must endorse the transaction and what criteria they must meet to do so.
- The endorsement policy is defined in the chaincode itself and can be customized to meet the needs of different applications. It is specified as a set of rules that define the minimum number of endorsements required for a transaction to be considered valid.

Hyperledger Fabric Network | Endorsement policy

- There are several types of endorsement policies that can be used in Hyperledger Fabric:

    1. **Signature-based endorsement policy:** This policy requires that a transaction be signed by a specified number of peers before it can be committed to the ledger.

    2. **Hash-based endorsement policy:** This policy requires that a transaction be hashed and the resulting hash value must match a specified value before the transaction can be committed to the ledger.

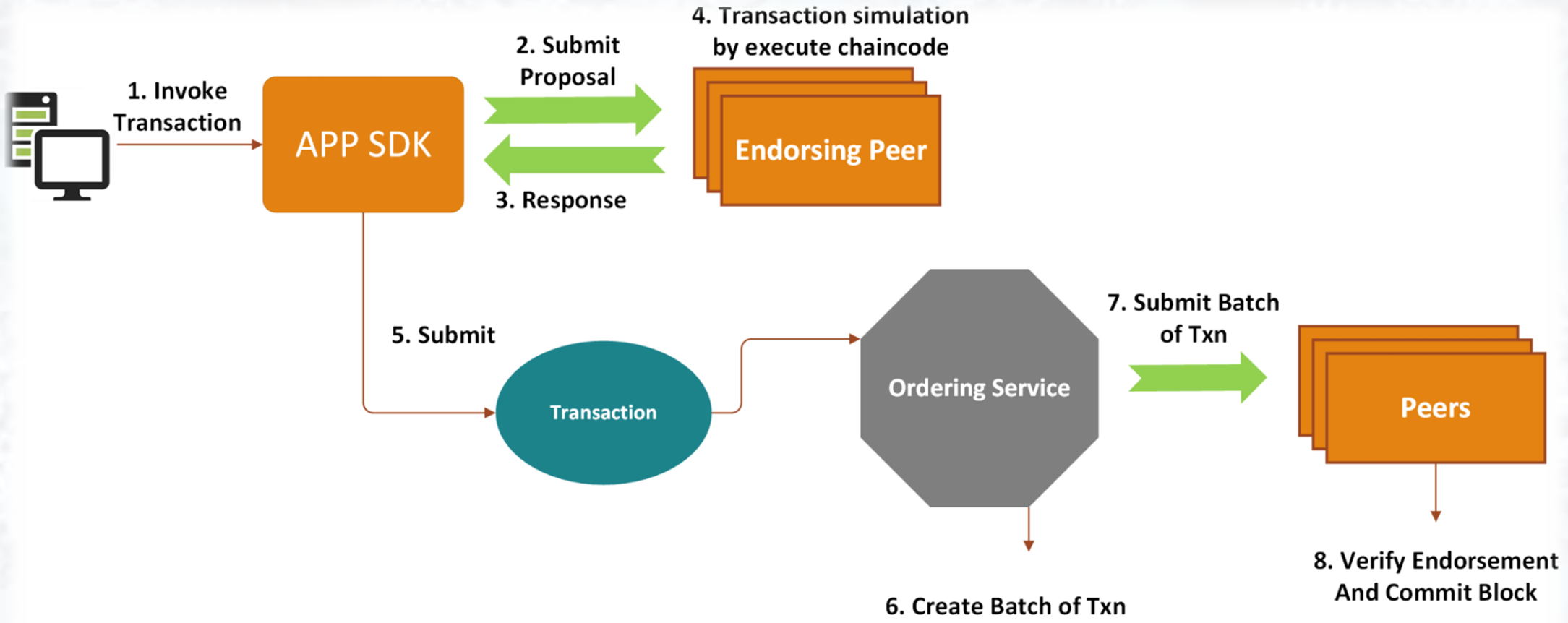
Hyperledger Fabric Network | Endorsement policy

- There are several types of endorsement policies that can be used in Hyperledger Fabric:

    3. **Implicit endorsement policy:** This policy assumes that all peers in the network are authorized to endorse transactions and therefore, all transactions are considered valid as long as they are signed by the submitting client.

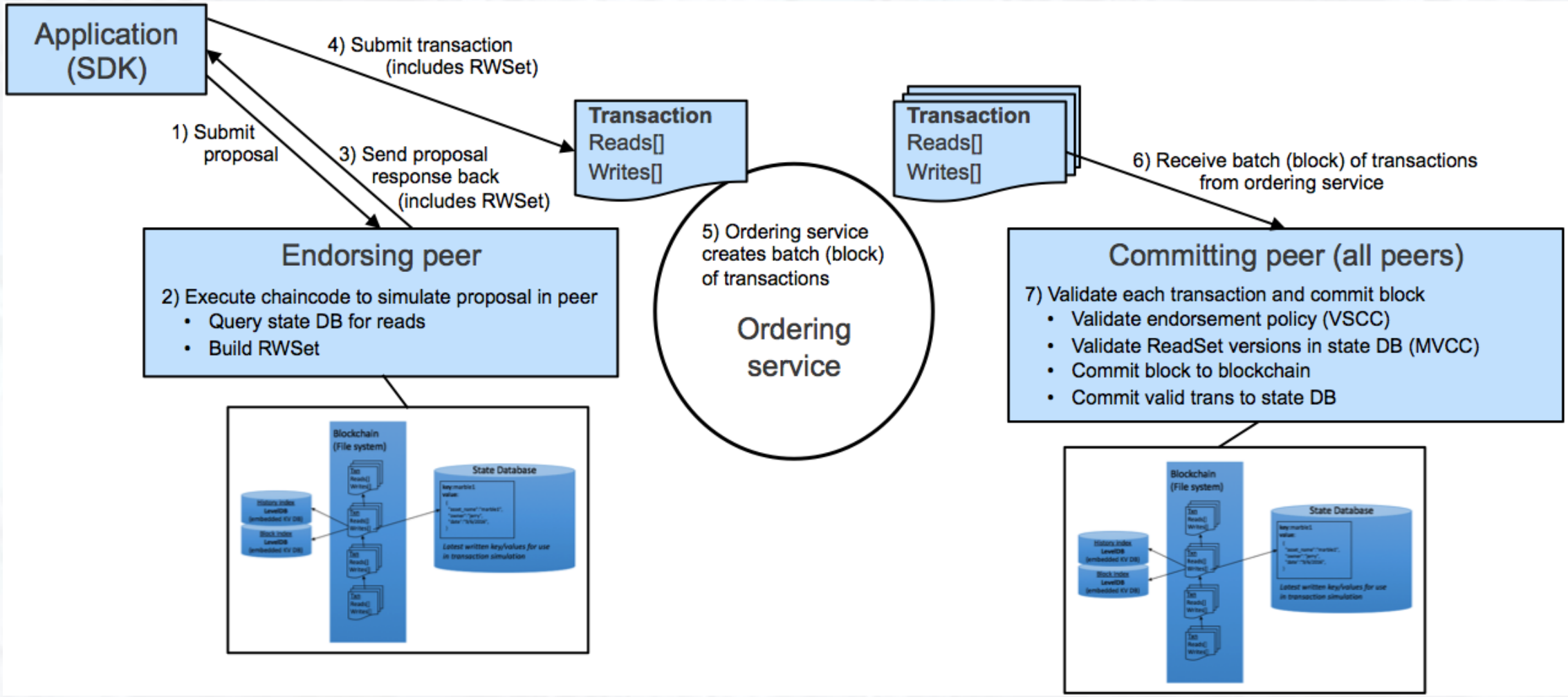
    4. **Distributed endorsement policy:** This policy distributes endorsement responsibility among multiple organizations or sub-groups within the network. For example, an endorsement policy might require that a transaction be endorsed by at least one peer from each organization in the network.

Hyperledger Fabric runtime architecture | Concept



Hyperledger Fabric transaction flow

Hyperledger Fabric runtime architecture | Concept



Hyperledger Fabric runtime architecture | Proposal

- The Hyperledger Fabric runtime architecture involves a number of different components working together to process transactions on the blockchain.
- The basic process can be broken down into four main stages:

1. **Transaction proposal:** The transaction proposal stage is where a client application initiates a transaction on the blockchain. The client application uses the Client SDK to send a transaction proposal to the endorsing peers in the network.

   - The transaction proposal contains the details of the transaction, including the chaincode that should be invoked and the parameters for that chaincode.

Hyperledger Fabric runtime architecture | Endorsement

- The Hyperledger Fabric runtime architecture involves a number of different components working together to process transactions on the blockchain.
- The basic process can be broken down into four main stages:

2. **Transaction endorsement:** Once the endorsing peers receive the transaction proposal, they evaluate the proposal to ensure that it meets the required criteria. This includes checking that the client has the necessary permissions to invoke the specified chaincode and that the parameters for the chaincode are valid.

   - If the transaction proposal passes the endorsement checks, the endorsing peers will sign the proposal and send it back to the client.

Hyperledger Fabric runtime architecture | Ordering

- The Hyperledger Fabric runtime architecture involves a number of different components working together to process transactions on the blockchain.
- The basic process can be broken down into four main stages:

3. **Transaction submitted to the ordering service:** After receiving the endorsements from the endorsing peers, the client can then submit the endorsed transaction to the ordering service.

   - The ordering service is responsible for ordering the transactions into a block, which will be added to the blockchain.

Hyperledger Fabric runtime architecture | Validation

- The Hyperledger Fabric runtime architecture involves a number of different components working together to process transactions on the blockchain.
- The basic process can be broken down into four main stages:

4. **Transaction validation:** Once the transaction has been ordered and added to the blockchain, it is then validated by the peer nodes in the network.

   - The validation process involves checking that the transaction is valid, that it has been endorsed by the required number of peers, and that the transaction has not already been included in a previous block.

Chaincode Lifecycle | Definition

- Chaincode Lifecycle refers to the process of installing, instantiating, and upgrading chaincode on a network.
- Each step of the lifecycle is important for ensuring that the network operates correctly and securely.
- The process involves multiple steps and requires coordination between the organizations that endorse transactions for the chaincode.
- Chaincode endorsement policies specify which organizations need to endorse a transaction for it to be considered valid, and can be changed during the chaincode lifecycle to adjust the network's security requirements.

Chaincode Lifecycle | Install

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric:

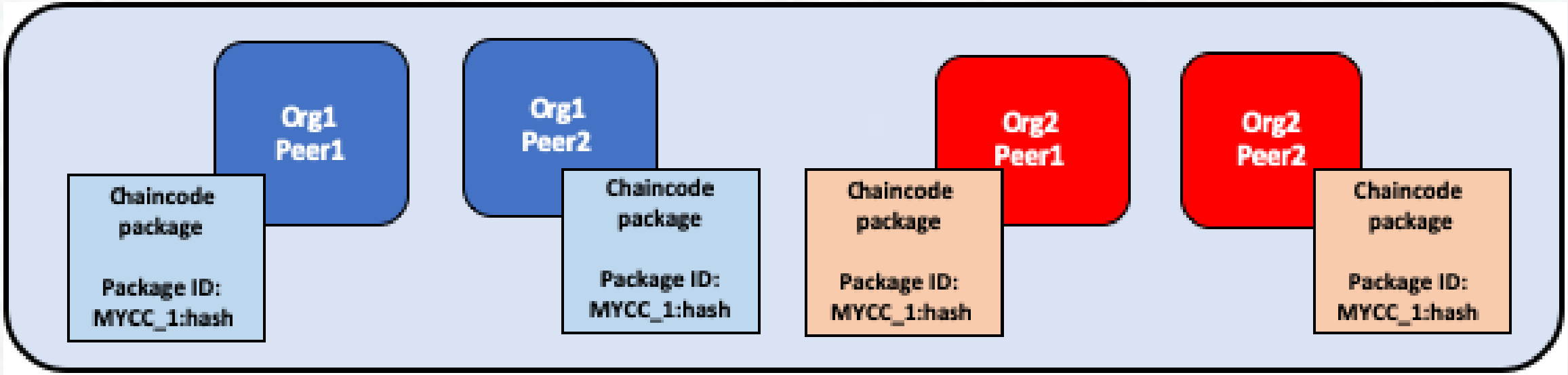
1. **Installing Chaincode:** The chaincode is typically written in a programming language such as Go or Node.js and packaged into a compressed file format that can be installed on peers.

- The package contains the chaincode source code, metadata, and dependencies. Peers can install multiple versions of the same chaincode. When a peer receives a request to execute a chaincode, it looks for the requested chaincode version that has been installed on that peer.

Chaincode Lifecycle | Install

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric:

1. **Installing Chaincode:** The chaincode is typically written in a programming language such as Go or Node.js and packaged into a compressed file format that can be installed on peers.



Chaincode Lifecycle | Approve

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric:

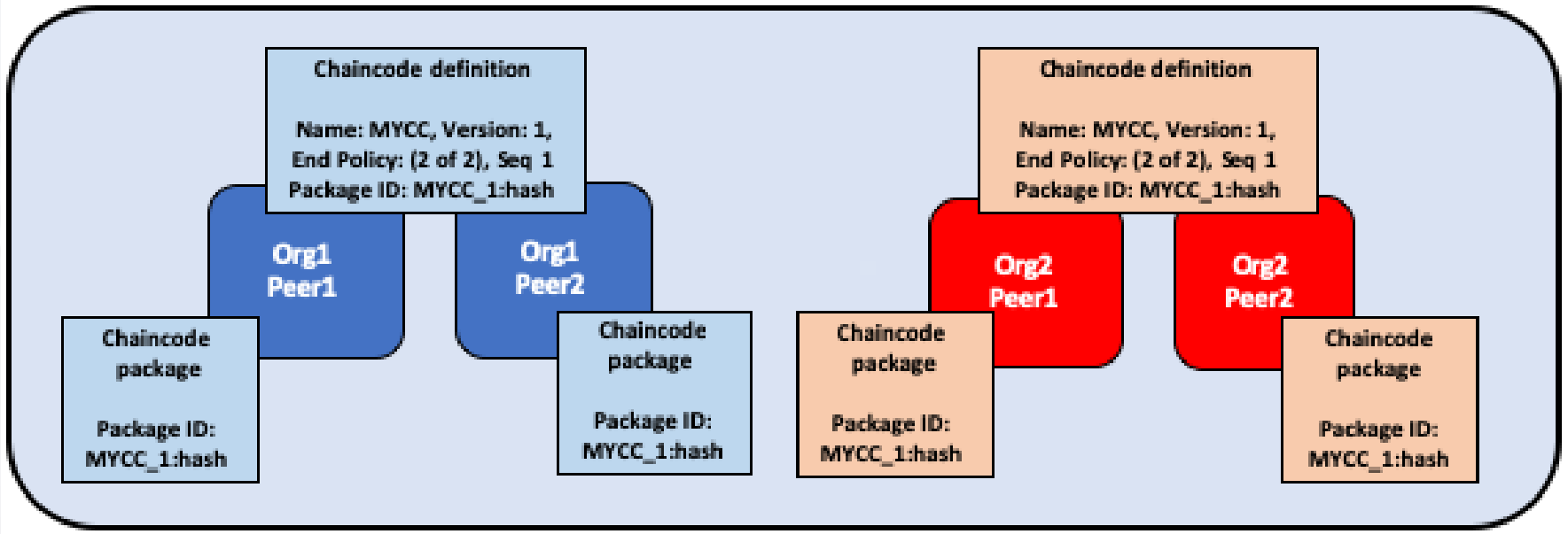
    2. **Approving Chaincode:** Before a chaincode can be committed to a channel, it needs to be approved by the organizations that will endorse transactions for that chaincode.

    - Each organization can set its own endorsement policy for a chaincode, specifying which peers can endorse transactions for that chaincode.
    - To approve a chaincode, an organization needs to run the "**`peer chaincode approve`**" command with the appropriate arguments, including the chaincode name, version, and endorsement policy.

Chaincode Lifecycle | Approve

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric:

2. **Approving Chaincode:** Before a chaincode can be committed to a channel, it needs to be approved by the organizations that will endorse transactions for that chaincode.



Chaincode Lifecycle | Commit

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric:

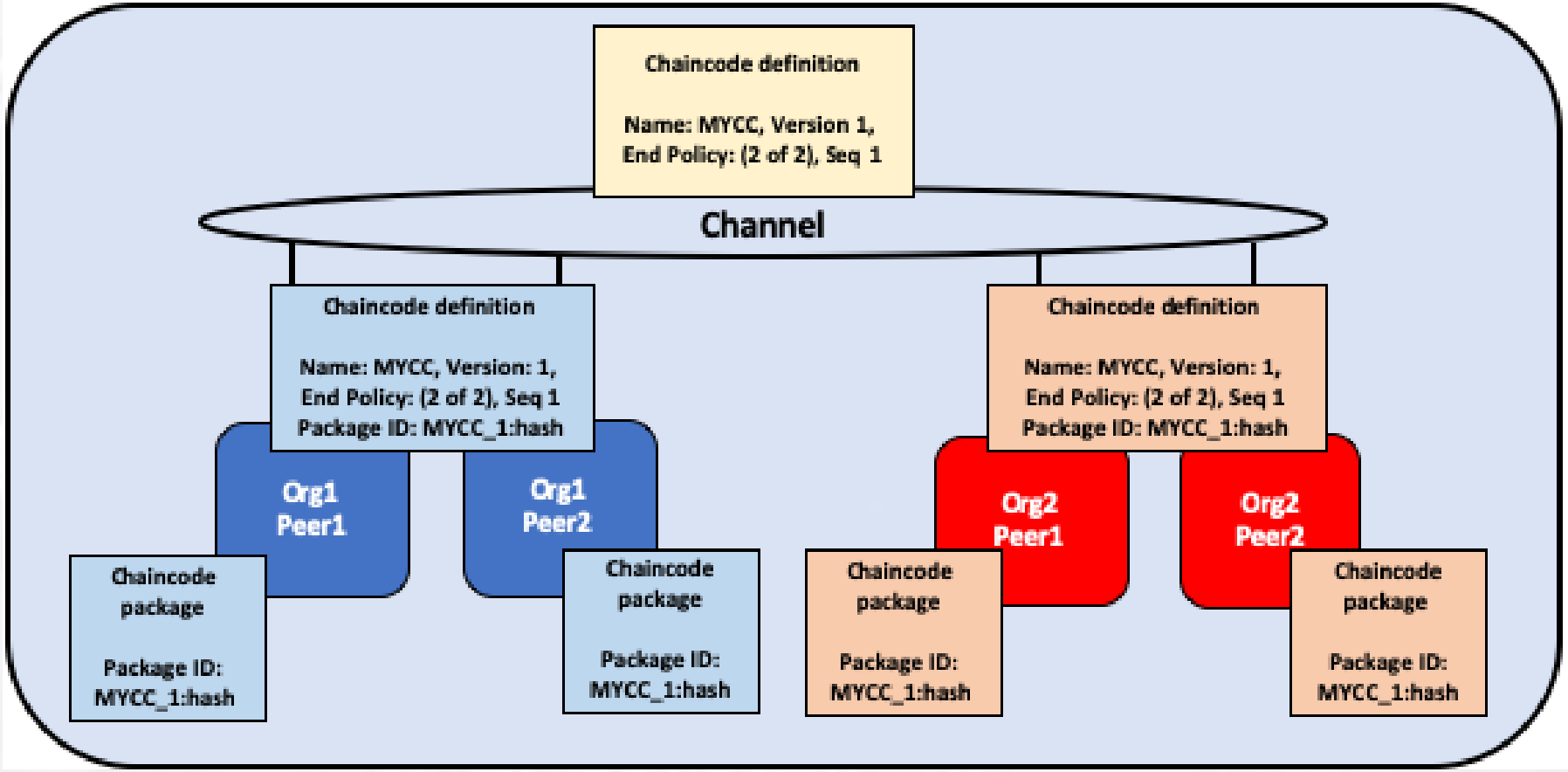
3. **Committing Chaincode:** After a chaincode is approved by all necessary organizations, it can be committed to a channel.

   - When a chaincode is committed to a channel, the chaincode metadata is stored in the channel's ledger, which includes the chaincode name, version, and endorsement policy.

   - Peers that belong to the channel can access the committed chaincode metadata.

Chaincode Lifecycle | Commit

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric: **3. Committing Chaincode:**



Chaincode Lifecycle | Start

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric:

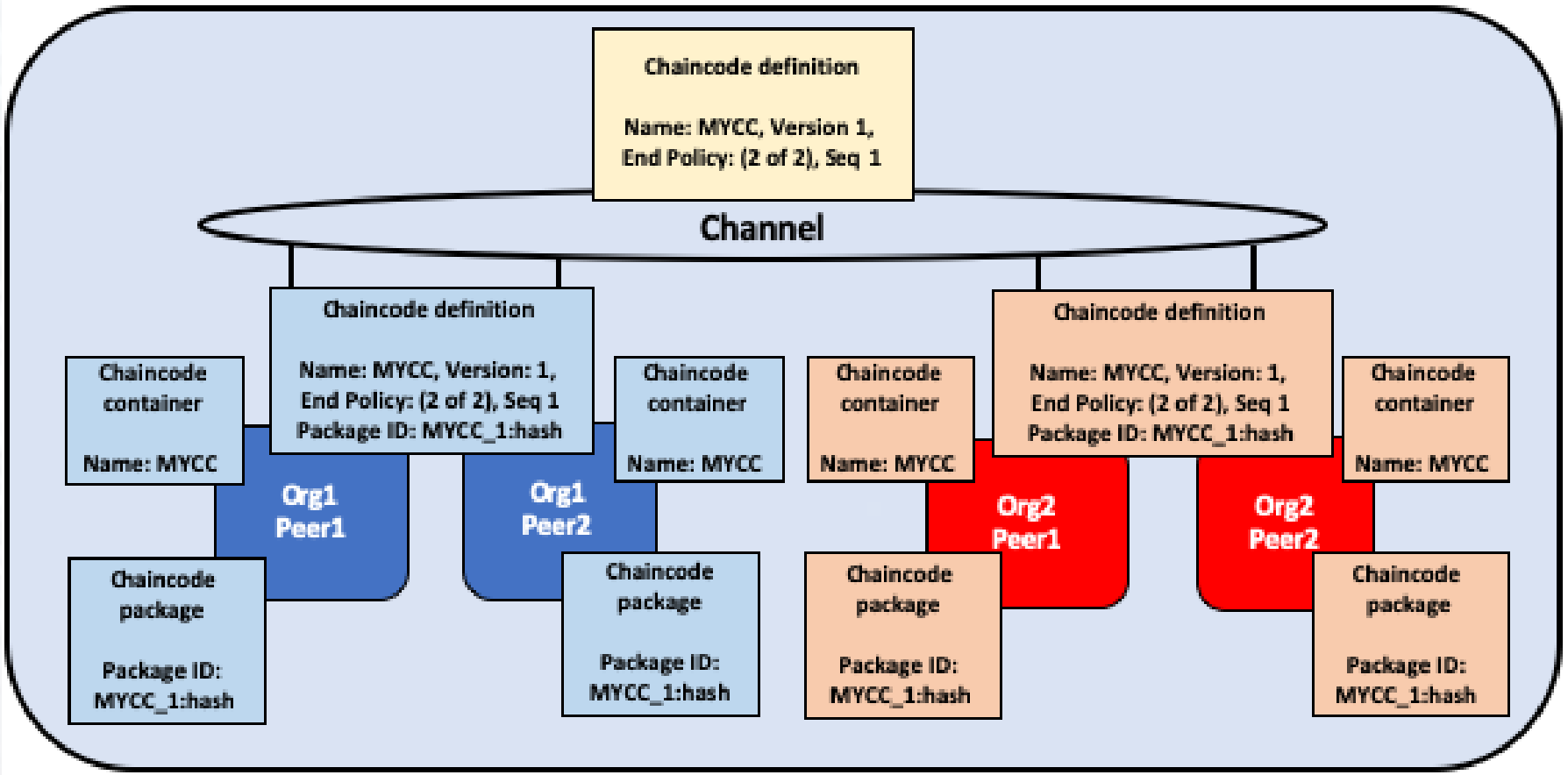
4. **Instantiating Chaincode:** Once the chaincode is committed to the channel, it can be instantiated on a specific peer.

   - The peer that instantiates the chaincode becomes the chaincode's owner and is responsible for executing the chaincode transactions. When a chaincode is instantiated, the peer creates an instance of the chaincode and assigns it a unique identifier.

   - The chaincode instance runs in a Docker container that is managed by the peer.

Chaincode Lifecycle | Start

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric: **4. Instantiating Chaincode:**



Chaincode Lifecycle | Upgrade

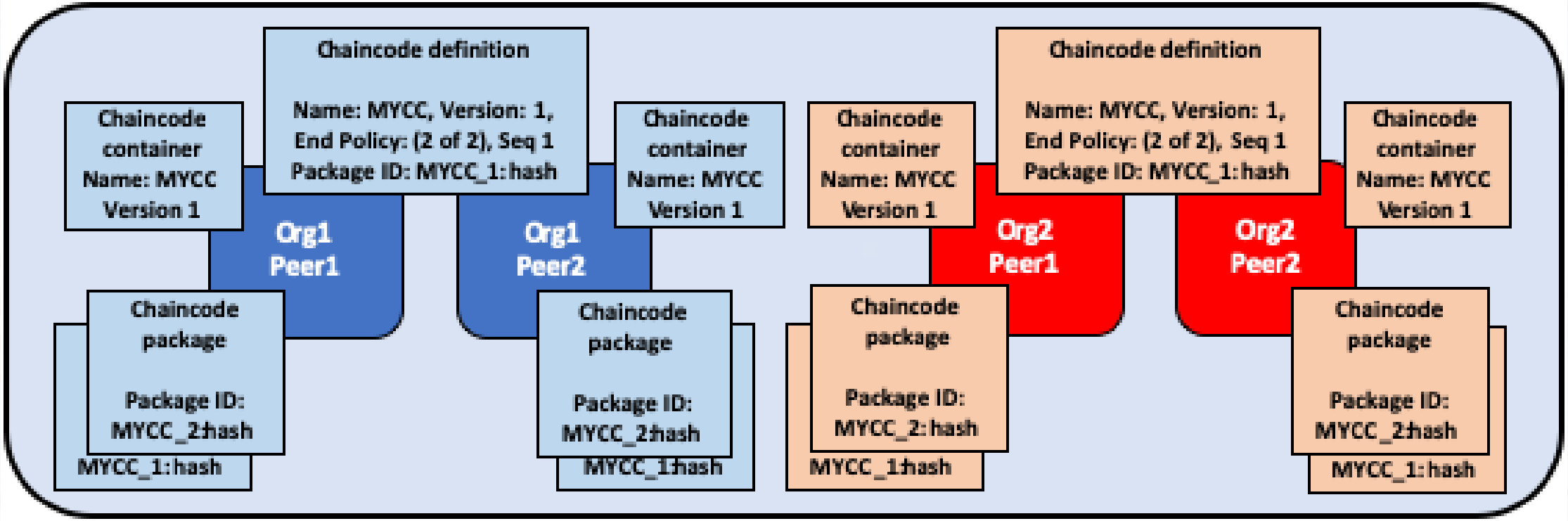
- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric:

    5. **Upgrading Chaincode:** If changes need to be made to the chaincode, a new version of the chaincode can be installed, approved, and then upgraded on the channel.

    - The upgrade process is similar to the commit process, except that the new version of the chaincode replaces the old version.
    - The upgrade process involves running the "**`peer chaincode upgrade`**" command with the appropriate arguments, including the new chaincode version, endorsement policy, and any initial arguments.

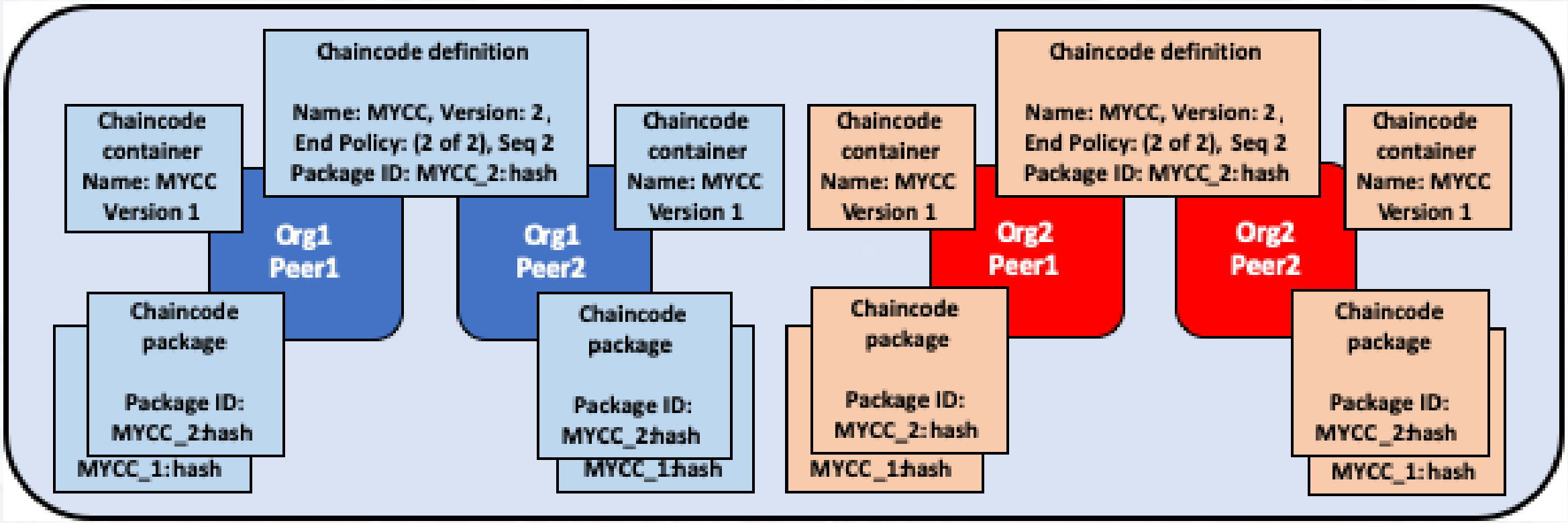
Chaincode Lifecycle | Upgrade - *Install*

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric: **5. Upgrading Chaincode: *Install.***



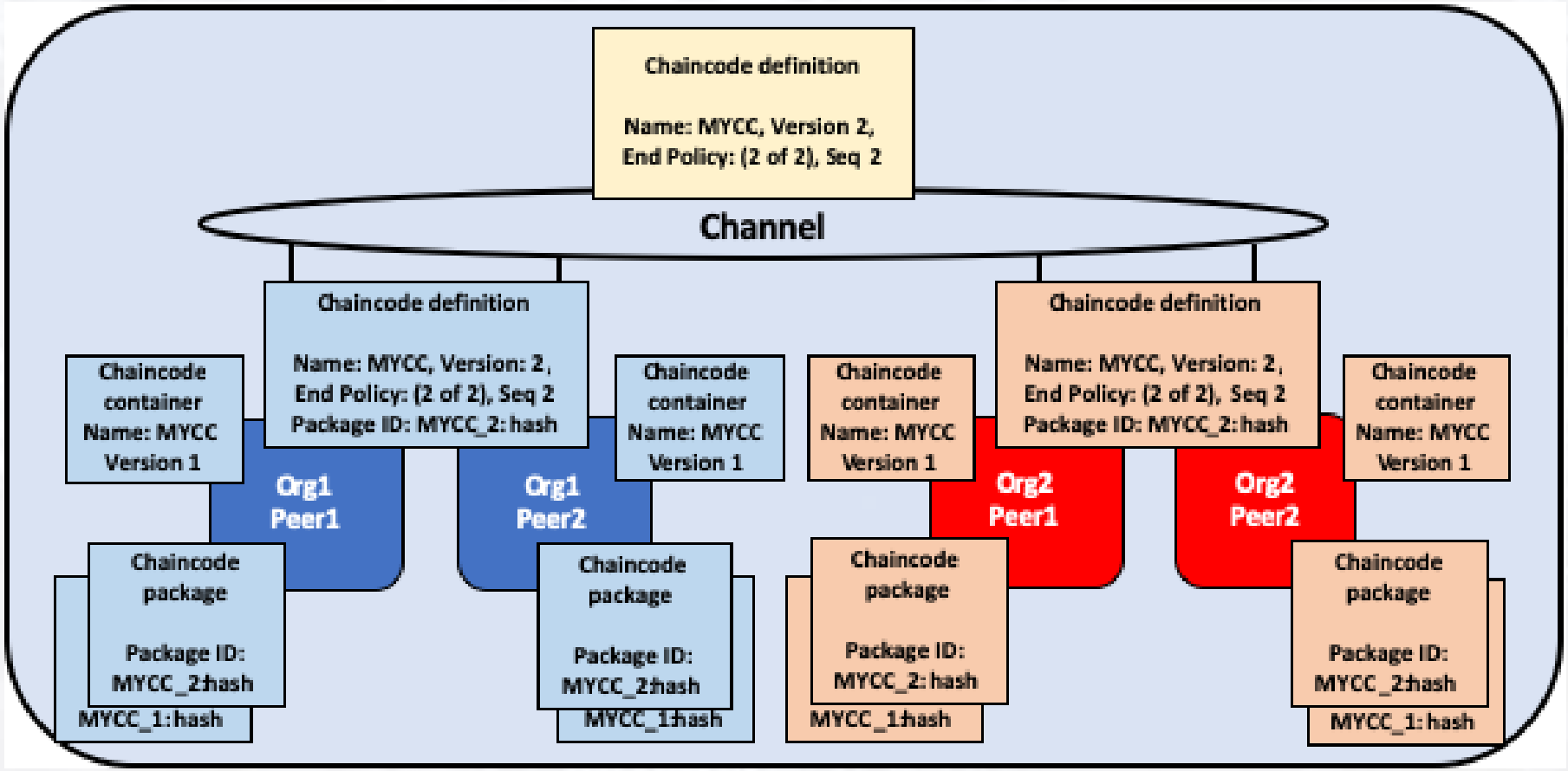
Chaincode Lifecycle | Upgrade - *Approve*

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric: **5. Upgrading Chaincode: *Approve.***



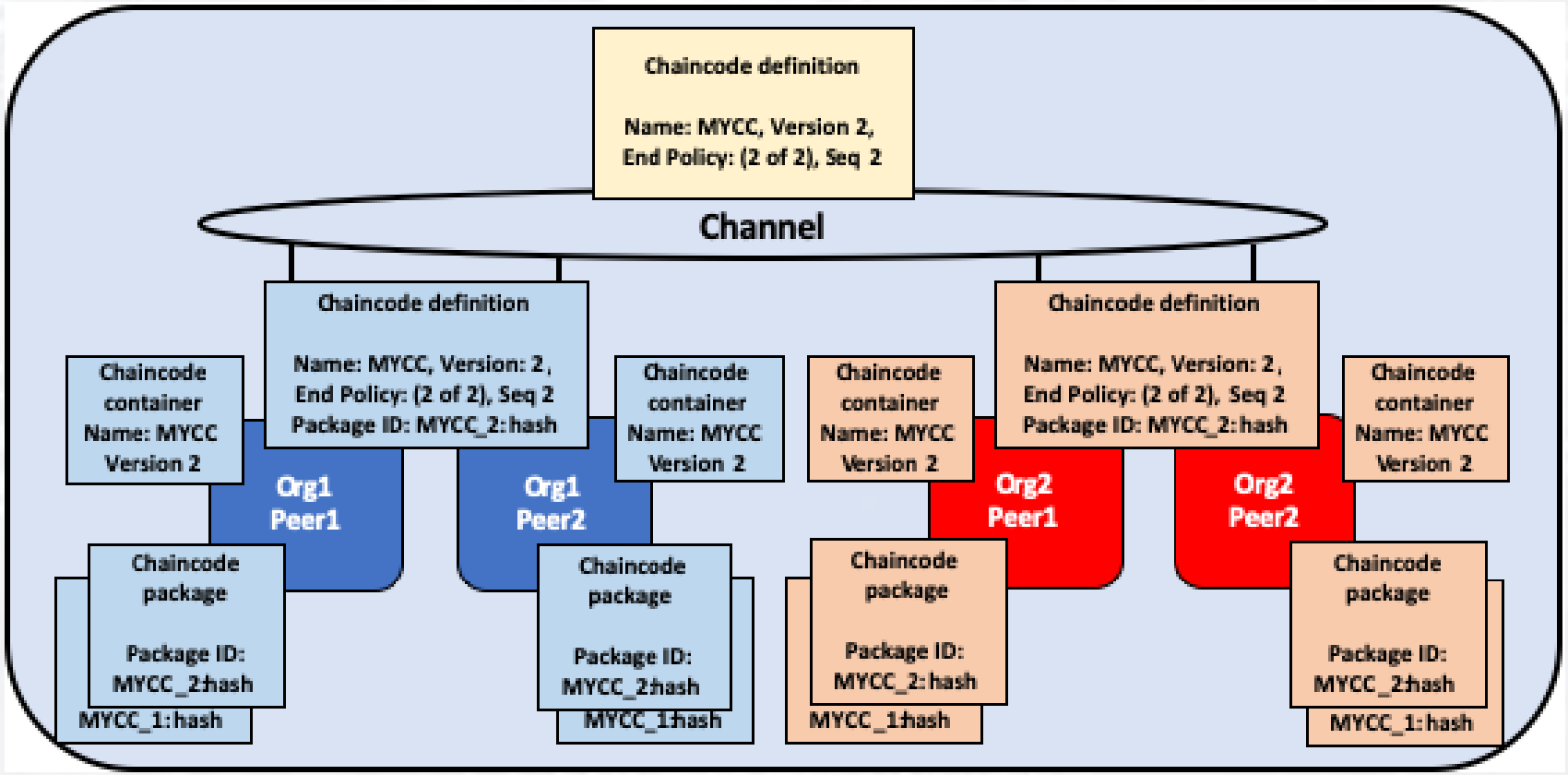
Chaincode Lifecycle | Upgrade - *Commit*

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric: **5. Upgrading Chaincode: *Commit.***



Chaincode Lifecycle | Upgrade - *Start*

- The following are the steps involved in the chaincode lifecycle in Hyperledger Fabric: **5. Upgrading Chaincode: *Instantiate.***



System chaincodes | Definition

- System chaincode is a special type of chaincode that is built into the Fabric framework and is used to perform tasks that are critical to the functioning of the network.
- It is responsible for tasks such as managing identities, performing endorsement and validation, and handling system events.
- System chaincodes are written in Go and are compiled into binaries that are distributed with the Fabric framework.
- These chaincodes are executed on peer nodes and provide services to user chaincodes.

System chaincodes | Lifecycle System Chaincode

- The commonly used system chaincodes in Hyperledger Fabric:

1. **Lifecycle System Chaincode (LSCC):** The LSCC manages the lifecycle of user chaincode. It is responsible for installing, instantiating, and upgrading user chaincode.

   - The LSCC interacts with the peer's ledger to store information about the user chaincode's state.

   - It also provides APIs for user chaincode to invoke during its lifecycle, such as querying the list of installed chaincodes or the version of a specific chaincode.

System chaincodes | Query System Chaincode

- The commonly used system chaincodes in Hyperledger Fabric:

  2. **Query System Chaincode (QSCC):** The QSCC provides APIs for querying the blockchain. It enables users to retrieve information about blocks, transactions, chaincodes, and more.

  - The QSCC is responsible for validating the queries and ensuring that the requested data is accessible to the user.
  - The QSCC is critical to the functioning of the network, as it provides the necessary visibility into the blockchain for users to understand the state of the network.

System chaincodes | Configuration System Chaincode

- The commonly used system chaincodes in Hyperledger Fabric:

  3. **Configuration System Chaincode (CSCC):** The CSCC manages the configuration of the network. It provides APIs for creating channels, joining peers to a channel, and updating channel policies.

  - The CSCC interacts with the ordering service to create or update channels and ensures that the configurations are consistent across the network.
  - The CSCC is responsible for enforcing the access control policies defined for the channels, such as ensuring that only authorized peers can join a channel.

System chaincodes | Endorsement System Chaincode

- The commonly used system chaincodes in Hyperledger Fabric:

4. **Endorsement System Chaincode (ESCC):** The ESCC is responsible for endorsing transactions during the execution of user chaincode. It verifies that the transaction meets the endorsement policy defined for the chaincode and generates an endorsement signature that is stored on the transaction.

- The ESCC is critical to the security of the network, as it ensures that transactions are validated before they are committed to the blockchain.

System chaincodes | Validation System Chaincode

- The commonly used system chaincodes in Hyperledger Fabric:

5. **Validation System Chaincode (VSCC):** The VSCC is responsible for validating transactions during the execution of user chaincode. It verifies that the transaction meets the validation policy defined for the chaincode and ensures that the transaction is consistent with the current state of the ledger.

   - The VSCC is critical to the integrity of the blockchain, as it ensures that only valid transactions are committed to the blockchain.

Introduction to Blockchain | Summary

►In this session, we discussed:

- Nodes in Hyperledger Fabric Network
- Endorsement policy
- Hyperledger Fabric runtime architecture
  - Transaction proposal (Client SDK)
  - Transaction endorsement
  - Transaction submitted to the ordering service
  - Transaction validation
- Chaincode lifecycle
- System chaincode


THANK
YOU