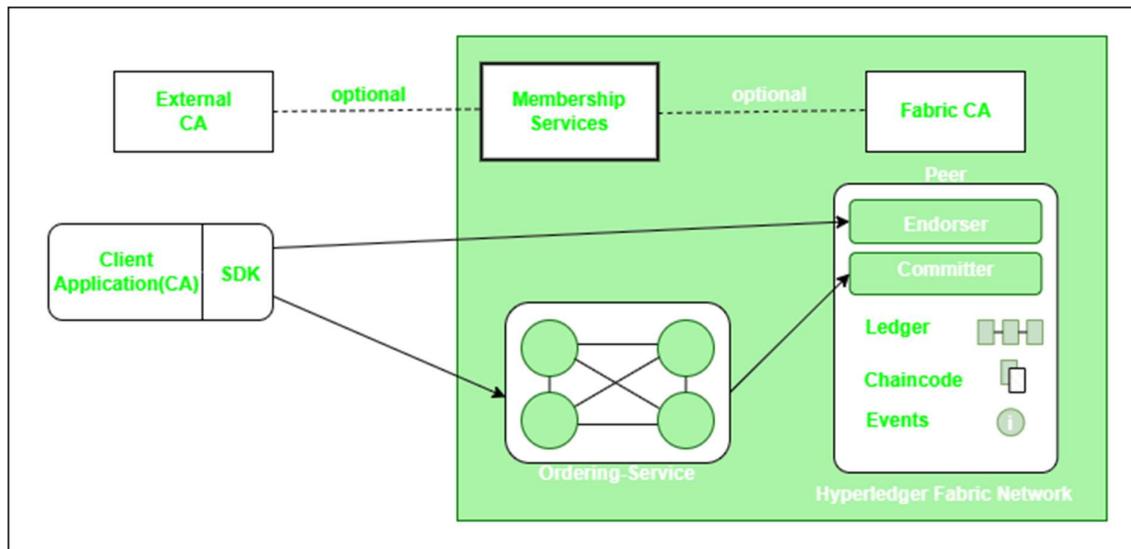


1. Draw and explain the architecture of hyper ledger fabric system and its components. (just for simplicity book chi diagram nhi use keli)



This Fig. shows how **different parts of Hyperledger Fabric work together** to create a secure, permissioned blockchain system.

1. Client Application (CA)

- This is the **user-side app** that wants to interact with the blockchain (e.g., make a payment, update a record).
- It uses an **SDK** (Software Development Kit) to send requests to the blockchain network.

2. Membership Services

- This handles **identity and access**.
- It makes sure **only trusted users** can use the network.
- It works with:
 - **Fabric CA** – Built-in certificate authority.
 - **External CA** – An optional third-party identity provider.

3. Ordering Service

- Think of it as the **traffic controller** of the network.
- It collects all the transaction requests from different users and **puts them in order**.
- This ensures all peers get the transactions in the **same sequence** (important for trust and accuracy).

4. Peers

These are the **main workers** in the network.

Each **peer** has:

- **Endorser**
 - It **runs the smart contract** (called **chaincode**) and checks if the transaction is valid.
 - If valid, it **signs (endorses)** the result and sends it back.
- **Committer**
 - It **receives the final ordered transaction** and **writes it to the ledger**.
 - This keeps the blockchain **updated**.

5. Chaincode (Smart Contract)

- This is the **business logic**.
- It defines **what to do** when a transaction is sent — like "transfer funds", "update marks", etc.

6. Ledger

- This stores **all past and current transactions**.
- It is a combination of:
 - **Blockchain** – Stores history (what happened).
 - **State database** – Stores current state (what's true now).

7. Events

- Used to **notify applications** when something happens in the network.
- For example, when a transaction is complete, the system can trigger an alert.

Flow Summary (Step-by-Step)

1. **Client App** sends a request via **SDK**.
2. **Endorsing Peers** run the smart contract (chaincode) and approve it.
3. The result goes to the **Ordering Service**, which arranges all transactions.
4. The **Committing Peers** record the final transactions into the **ledger**.
5. The **Membership Services** make sure only **verified users** can participate.

2. Hyper ledger? And features of a hyper ledger Blockchain

- **Hyperledger** is an **open-source project** started by the **Linux Foundation** in 2015.
- It is not a single blockchain but a **collection of blockchain frameworks and tools** designed especially for **business (enterprise) use**.
- Think of it as a platform that lets companies **build their own private and secure blockchain systems**.

Features: -

- Only authorized users are allowed to access and participate in the blockchain network, making it more secure and trusted.
- The system is flexible, allowing developers to select and plug in only the components they need, like databases, consensus mechanisms, etc.
- Hyperledger can process a large number of transactions quickly, making it suitable for business environments.
- Transactions can be kept private between selected parties, so sensitive data is not visible to everyone on the network.
- Business logic and rules can be automated using smart contracts, which execute automatically when conditions are met.
- It supports a large number of users and transactions without reducing performance, making it highly scalable.
- There is no need for mining or any cryptocurrency to run the system, which reduces cost and energy usage.
- Each user has a unique digital identity, helping in tracking actions and managing access securely.

3. Difference between Ethereum and Hyper ledger fabric Blockchain

| No. | Feature | Ethereum | Hyperledger Fabric |
|-----|-------------------------|-----------------------------|-----------------------------------|
| 1 | Type | Public blockchain | Private (permissioned) blockchain |
| 2 | Access | Open to anyone | Restricted to authorized users |
| 3 | Cryptocurrency | Uses Ether (ETH) | No built-in cryptocurrency |
| 4 | Consensus Mechanism | Proof of Stake (PoS) | Pluggable (e.g., Raft, Kafka) |
| 5 | Speed | Slower due to public access | Faster due to controlled access |
| 6 | Smart Contract Language | Solidity | Go, Java, JavaScript |

| | | | |
|-----------|----------------------------|---------------------------------------|---|
| 7 | Use Case Focus | Decentralized apps & DeFi | Enterprise business applications |
| 8 | Transaction Cost | Requires gas fees | Usually, no transaction fees |
| 9 | Transparency | Fully transparent to all participants | Data visible only to authorized parties |
| 10 | Identity Management | Anonymous or pseudonymous users | Strong identity verification system |

4. Working of hyper ledger fabric

Step by Step: -

- **Network Setup:**
Organizations join the network by setting up peer nodes and assigning digital identities to users.
- **Smart Contract (Chaincode):**
Business logic is written in chaincode, which defines how transactions should work.
- **Channel Creation (Optional):**
Private channels are made so only selected participants can see certain data.
- **Transaction Proposal:**
A user sends a request (proposal) to do something, like updating a record.
- **Endorsement:**
Selected peers simulate the transaction and approve it if it follows the rules.
- **Ordering Service:**
Approved transactions are collected and ordered into a block.
- **Validation:**
All peers check the block to make sure the transactions are valid.
- **Commit to Ledger:**
Valid transactions are added to the blockchain and the world state is updated.
- **User Notification:**
The user is told whether the transaction was successful.

Example: -

- A mango farm, shipping company, and supermarket are part of a Hyperledger Fabric network.
- The farm harvests mangoes and sends a transaction proposal to record the batch info.
- The shipping company and supermarket endorse (approve) the transaction.
- The transaction is sent to the ordering service and added to a block.
- All peers validate and add the block to their ledgers.
- The mango shipment is now traceable by all authorized members in the network.

5. Hyper ledger technology and hyper ledger fabric.

| Feature | Hyperledger Technology | Hyperledger Fabric |
|---------------------|--|---|
| Definition | Open-source umbrella project for enterprise blockchain tools | A specific blockchain framework under Hyperledger |
| Purpose | Supports multiple blockchain projects for different needs | Designed for permissioned blockchain networks |
| Number of Projects | Includes many projects like Fabric, Sawtooth, Besu, Indy, etc. | One single project within Hyperledger |
| Network Type | Focuses on permissioned (private) blockchains | Permissioned blockchain framework |
| Consensus Mechanism | Varies by project; supports pluggable consensus | Modular consensus with no mining |
| Smart Contracts | Depends on the project | Uses chaincode in Go, Java, or JavaScript |
| Use Cases | Broad industries like finance, supply chain, healthcare | Enterprise apps in finance, supply chain, healthcare |
| Cryptocurrency | Usually no built-in cryptocurrency | No cryptocurrency or tokens |
| Modularity | Provides modular tools and frameworks for customization | Modular architecture allows plug-and-play components |
| Governance | Governed by Linux Foundation with multiple collaborators | Managed by specific working groups within Hyperledger |

6. Benefits of hyper ledger fabric.

1. Hyperledger Fabric is a permissioned blockchain, so only trusted users can access the network.
2. It has a modular architecture, allowing customization of components like consensus and membership.
3. Private channels enable confidential transactions between specific participants.
4. It does not require cryptocurrency or mining, making it efficient and low-cost.

5. The network is scalable and fast because it separates transaction endorsement, ordering, and validation.
6. Supports smart contracts (chaincode) written in popular programming languages like Go and JavaScript.
7. Strong identity management verifies the identity of all participants securely.
8. Offers enterprise-grade security suitable for handling sensitive business data.
9. Supports interoperability, allowing integration with existing systems and other blockchains.
10. Provides detailed audit trails, ensuring transparency and accountability in transactions.