**Developer Tools**

• Hyperledger Fabric SDK libraries

• Java

**API’s**

• User details are recorded through the User Portal Application which populate Blocks (i.e. —user details placed into block form via json format)

• Member Applications

• Use batching for efficiency

**Channel**

•At this point, we require only one channel within our Network. Permission access is required for members of the Network.

•More channels could be added later in case further partitions are needed among members.

**Consortium**

• We have five consortium members: 1) Corporation Authority, 2) Food and Safety Authority, 3) Economic Zone Authority, 4) Bank, 5) Order Service (…kind of like a traffic controller with pre-approved logic) — UAE Trade Connect

**Data**

• Data Structure stored as Hash Sets {Key: Value} — Provides constant-time query methods, avoids duplications

• Local Ledger kept by each participant giving us a ‘distributed’ structure (built in data redundancy as well).

• Possibility for private data collections in case some transactions or data is deemed ‘sensitive’ in nature.

• Alternatively, data fields can be hidden using hashing.

• Global data store which is independent of local ledgers. Holds the ‘state’.

• Immutable Master Blockchain ledger with SHA-256 hash

**Chain-code**

• Business logic in code form — [For example, based on (Corporation Authority) and (Food and Safety Authority) approvals, Economic Zone Authority approves the request].

**Communication**

• Multi-casting/Transmitting data between nodes via sockets

**Consensus**

1 Ordering of transactions

2 Validation of transactions

• Suggest using the Raft Protocol

• Uses a (leader and follower) model, in which a leader is dynamically elected among the ordering nodes in a channel. The leader replicates messages to the follower nodes.

• Crash fault tolerant” (CFT) with a minimum quorum of three nodes in channel

**Identity**

• Authentication of certificates and public keys which are initially provided by the MSP (Membership Service Provider)

**Risk Analysis**

• Monte Carlo Simulations for testing to produce a ranges of acceptable metrics

**Performance**

• A trade-off between speed and scalability because nodes transfer the messages.

Fewer nodes mean fast speed for reaching consensus. On the other hand, scaling to more nodes means slower speed for reaching consensus.

• Low-latency finality