Below files are needed for the project

1. configtx.yaml
2. cypto-confifg1.yaml
3. docker-compose-cli.yaml
4. docker-compose-couch.yaml
5. docker-compose-e2e-template.yaml

Note: Update the CA certificate in this file

Cytpto\_config ---- peerorgranization—org1.batch3.com---ca

Same for org2

Under base folder

1. docker-compose-base.yaml
2. Peer-base.yaml
3. .Env

Steps to Execute the project :

Reference document: <https://hyperledger-fabric.readthedocs.io/en/release-1.4/build_network.html>

Step1:

../bin/cryptogen generate --config=./crypto-config.yaml

Step 2:

export FABRIC\_CFG\_PATH=$PWD

Step 3:

Mkdir channel-artifacts

../bin/configtxgen -profile TwoOrgsOrdererGenesis -channelID byfn-sys-channel -outputBlock ./channel-artifacts/genesis.block

Step4:

export CHANNEL\_NAME=mychannel && ../bin/configtxgen -profile TwoOrgsChannel -outputCreateChannelTx ./channel-artifacts/channel.tx -channelID mychannel

Step5:

../bin/configtxgen -profile TwoOrgsChannel -outputAnchorPeersUpdate ./channel-artifacts/Org1MSPanchors.tx -channelID mychannel -asOrg Org1MSP

Step6:

../bin/configtxgen -profile TwoOrgsChannel -outputAnchorPeersUpdate ./channel-artifacts/Org2MSPanchors.tx -channelID mychannel -asOrg Org2MSP

Step 7:

docker rm -f $(docker ps -aq)

Step 8:

docker rmi -f $(docker images -q)

docker network prune

Step9:

docker ps -a

Step 10:

docker-compose -f docker-compose-e2e-template.yaml -f docker-compose-cli.yaml -f docker-compose-couch.yaml up -d

Step11:

docker ps -a

Step12:

docker exec -it cli bash

step 13:

CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.batch3.com/users/Admin@org1.batch3.com/msp

CORE\_PEER\_ADDRESS=bharat.org1.batch3.com:7051

CORE\_PEER\_LOCALMSPID=Org1MSP

CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.batch3.com/peers/bharat.org1.batch3.com/tls/ca.crt

export CHANNEL\_NAME=mychannel

Step 14: updating anchor peer- bharat of org1

peer channel create -o orderer.batch3.com:7050 -c mychannel -f ./channel-artifacts/channel.tx --tls --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/batch3.com/orderers/orderer.batch3.com/msp/tlscacerts/tlsca.batch3.com-cert.pem

Step15: Joining channel – bharat of org1

peer channel join -b mychannel.block

**step 16:** Joining channel – afroz of org1

CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.batch3.com/users/Admin@org1.batch3.com/msp

CORE\_PEER\_ADDRESS=afroz.org1.batch3.com:8051

CORE\_PEER\_LOCALMSPID=Org1MSP

CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.batch3.com/peers/bharat.org1.batch3.com/tls/ca.crt

peer channel join -b mychannel.block

Step 17:

CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.batch3.com/users/Admin@org2.batch3.com/msp

CORE\_PEER\_ADDRESS=kumar.org2.batch3.com:9051

CORE\_PEER\_LOCALMSPID=Org2MSP

CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.batch3.com/peers/kumar.org2.batch3.com/tls/ca.crt

Step 18: updating anchor peer- kumar of org2

peer channel update -o orderer.batch3.com:7050 -c mychannel -f ./channel-artifacts/Org2MSPanchors.tx --tls --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/batch3.com/orderers/orderer.batch3.com/msp/tlscacerts/tlsca.batch3.com-cert.pem

Step 19: Joining channel

peer channel join -b mychannel.block

step 20: Vazir joining channel – org2

CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.batch3.com/users/Admin@org2.batch3.com/msp

CORE\_PEER\_ADDRESS=vazir.org2.batch3.com:10051

CORE\_PEER\_LOCALMSPID=Org2MSP

CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.batch3.com/peers/vazir.org2.batch3.com/tls/ca.cr

peer channel join -b mychannel.block

step 21: Installing chain code on org1- bharat

CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.batch3.com/users/Admin@org1.batch3.com/msp

CORE\_PEER\_ADDRESS=bharat.org1.batch3.com:7051

CORE\_PEER\_LOCALMSPID=Org1MSP

CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.batch3.com/peers/bharat.org1.batch3.com/tls/ca.crt

peer chaincode install -n mycc -v 1.0 -p github.com/chaincode/chaincode\_example02/go

step 22: Installing chain code on org2- Kumar

CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.batch3.com/users/Admin@org2.batch3.com/msp

CORE\_PEER\_ADDRESS=kumar.org2.batch3.com:9051

CORE\_PEER\_LOCALMSPID=Org2MSP

CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.batch3.com/peers/kumar.org2.batch3.com/tls/ca.crt

peer chaincode install -n mycc -v 1.0 -p github.com/chaincode/chaincode\_example02/go

### Step 23: Instantiate Chaincode

CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.batch3.com/users/Admin@org1.batch3.com/msp

CORE\_PEER\_ADDRESS=bharat.org1.batch3.com:7051

CORE\_PEER\_LOCALMSPID=Org1MSP

CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.batch3.com/peers/bharat.org1.batch3.com/tls/ca.crt

peer chaincode instantiate -o orderer.batch3.com:7050 --tls --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/batch3.com/orderers/orderer.batch3.com/msp/tlscacerts/tlsca.batch3.com-cert.pem -C mychannel -n mycc -v 1.0 -c '{"Args":["init","a", "100", "b","200"]}' -P "AND ('Org1MSP.peer','Org2MSP.peer')"

### Step 24: Query

peer chaincode query -C mychannel -n mycc -c '{"Args":["query”,”a”]}’

### Step 25: Invoke

peer chaincode invoke -o orderer.batch3.com:7050 --tls true --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/batch3.com/orderers/orderer.batch3.com/msp/tlscacerts/tlsca.batch3.com-cert.pem -C mychannel -n mycc --peerAddresses bharat.org1.batch3.com:7051 --tlsRootCertFiles /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.batch3.com/peers/bharat.org1.batch3.com/tls/ca.crt --peerAddresses kumar.org2.batch3.com:9051 --tlsRootCertFiles /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.batch3.com/peers/kumar.org2.batch3.com/tls/ca.crt -c '{"Args":["invoke","a","b","10"]}'

### Step 26 : Query

peer chaincode query -C mychannel -n mycc -c '{"Args":["query","a"]}'

peer chaincode query -C mychannel -n mycc -c '{"Args":["query","b"]}'

Step 27:

Login into the Kumar org2 and query – you will get same result

CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.batch3.com/users/Admin@org2.batch3.com/msp

CORE\_PEER\_ADDRESS=kumar.org2.batch3.com:9051

CORE\_PEER\_LOCALMSPID=Org2MSP

CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.batch3.com/peers/kumar.org2.batch3.com/tls/ca.crt

### Step 28 : Query

peer chaincode query -C mychannel -n mycc -c '{"Args":["query","a"]}'

Step 29: Install the chaincode on other peers – Afroz of org1 and vazir of org2 and query u will get same result