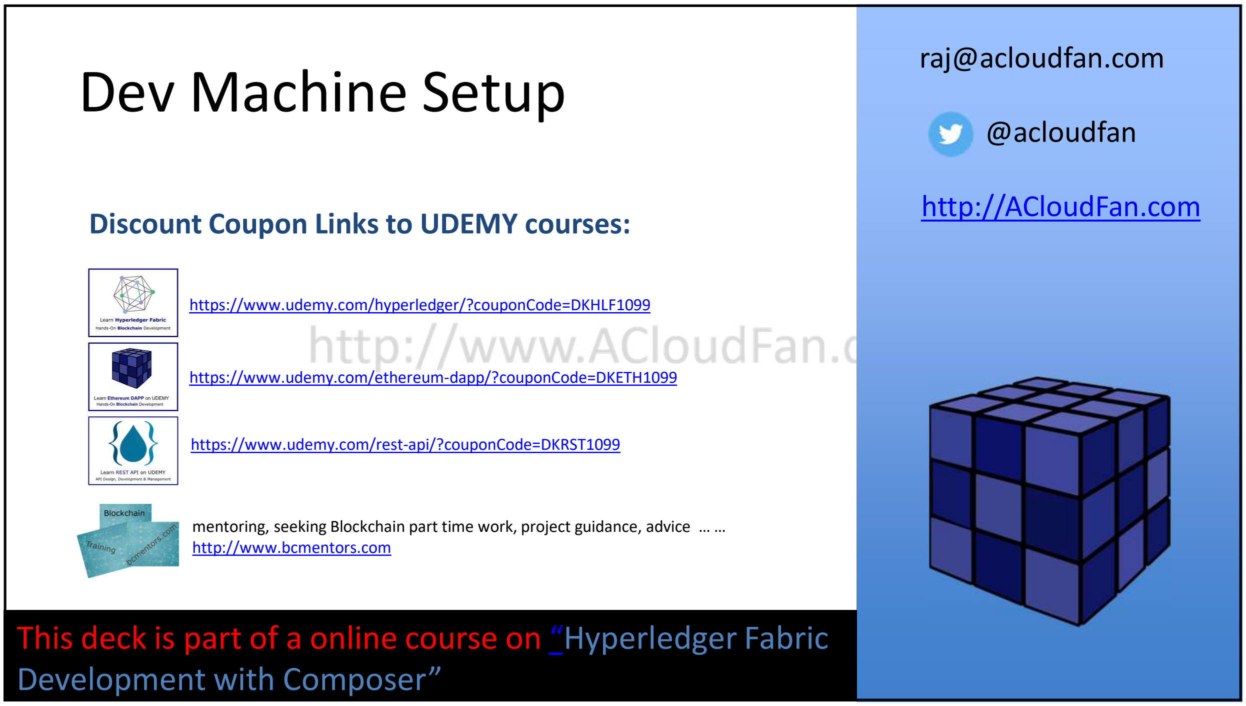


Dev Machine Setup

raj@acloudfan.com

@acloudfan

http://ACloudFan.com

Discount Coupon Links to UDEMY courses:

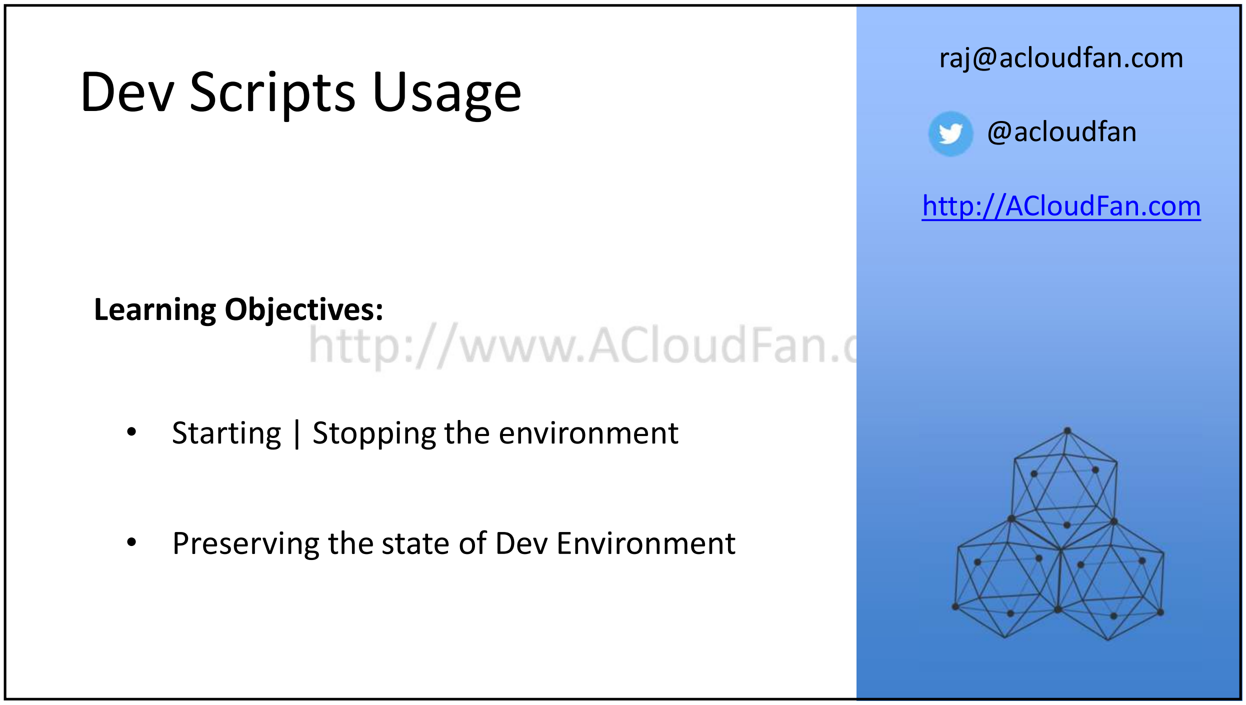
https://www.udemy.com/hyperledger/?couponCode=DKHLF1099

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"



Dev Scripts Usage

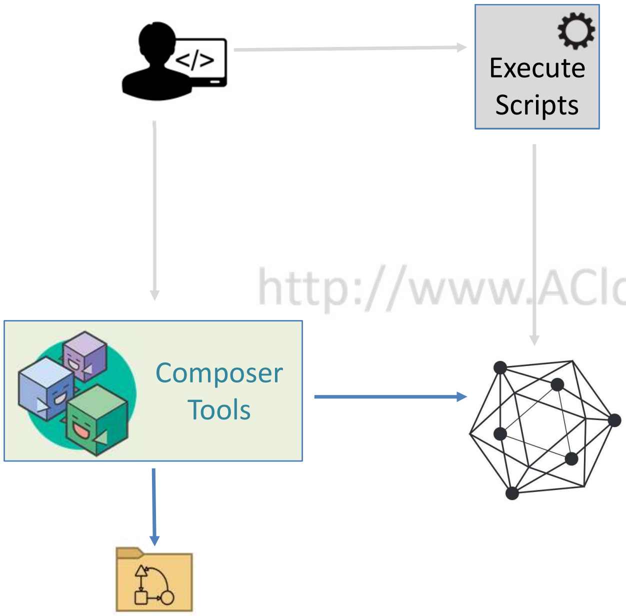
raj@acloudfan.com

@acloudfan

http://ACloudFan.com

Learning Objectives:

- Starting | Stopping the environment
- Preserving the state of Dev Environment



Launch Dev Environment

Shutdown Dev Environment

Suspend/Restart Dev Environment

LAUNCH Dev Environment

To launch the Fabric Dev Environment

```
$./startFabric.sh
```

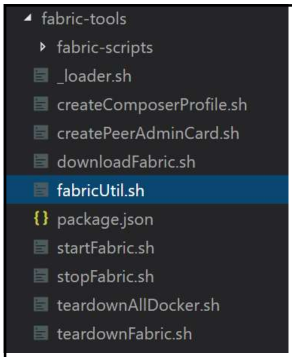
- Kills & Removes running containers
- All deployed applications are REMOVED
- Re-deployment of application will be needed

Shutdown Dev Environment

To shutdown the Fabric Dev Environment

```
$./stopFabric.sh
```

- Kills & Removes running containers
- All deployed applications are REMOVED
- Re-deployment of application will be needed



```
fabric-tools
  fabric-scripts
  _loader.sh
  createComposerProfile.sh
  createPeerAdminCard.sh
  downloadFabric.sh
  fabricUtil.sh
  package.json
  startFabric.sh
  stopFabric.sh
  teardownAllDocker.sh
  teardownFabric.sh
```

← Do you have this script under fabric-tools folder ?

If NOT:

Copy it from Link provided in the resources

```
chmod 755 fabricUtil.sh
```

Non-destructive Restarts of Dev Environment

To shutdown Dev Environment without losing deployed Apps

- First time launch

```
$./startFabric.sh
```

<< Deploy apps >>

```
$./fabricUtil.sh    stop

$./fabricUtil.sh    start
```

Dev Environment Topology

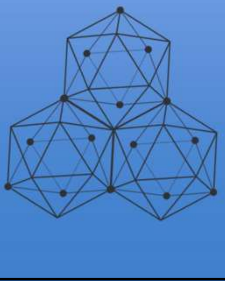
raj@acloudfan.com

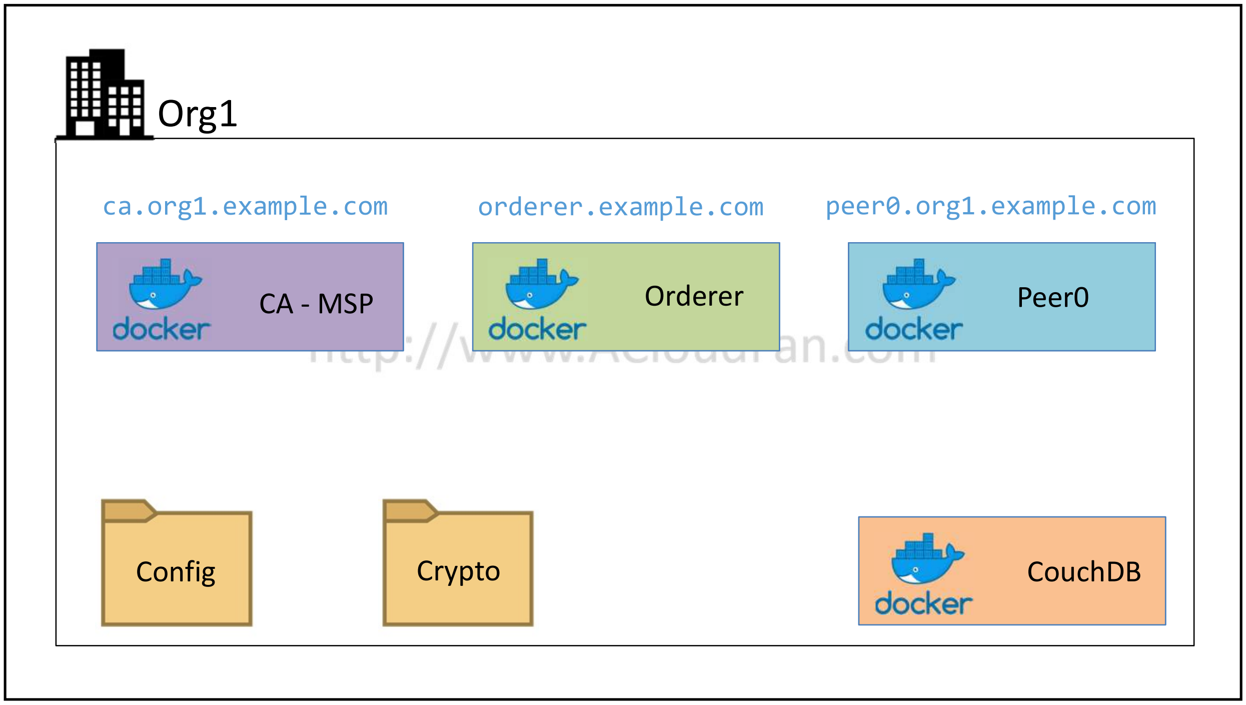
@acloudfan

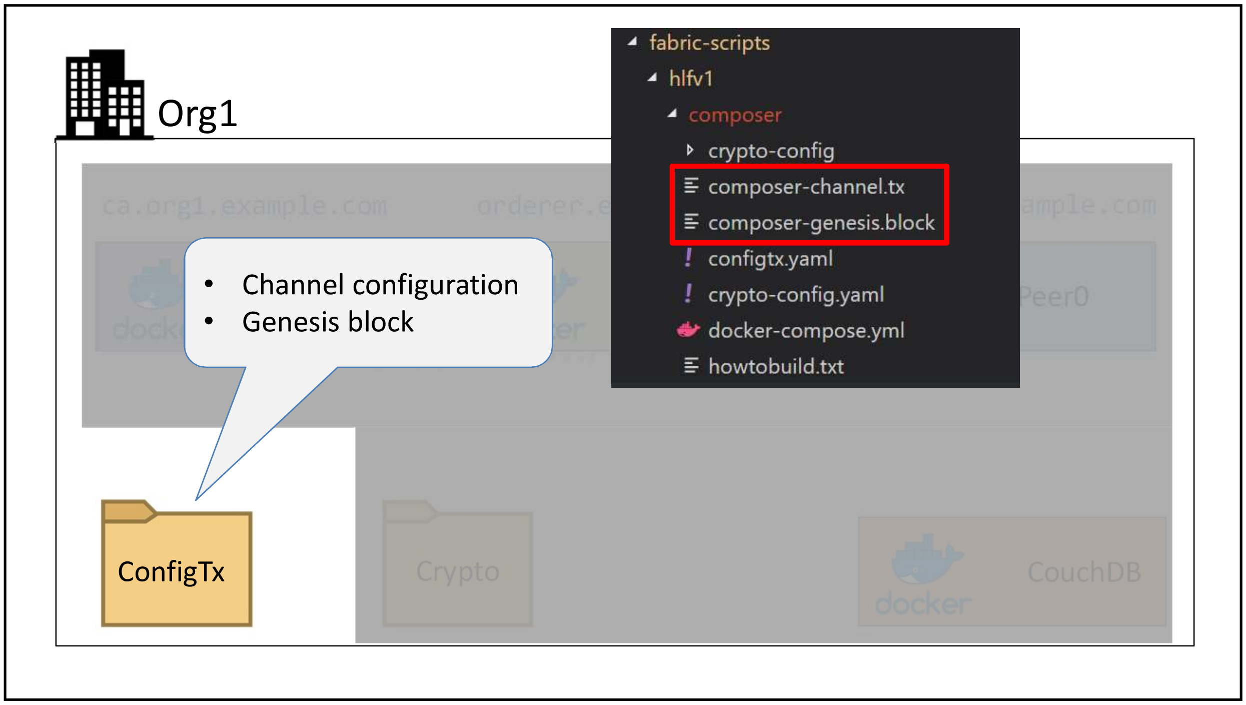
http://ACloudFan.com

Learning Objectives:

- Overview of the setup

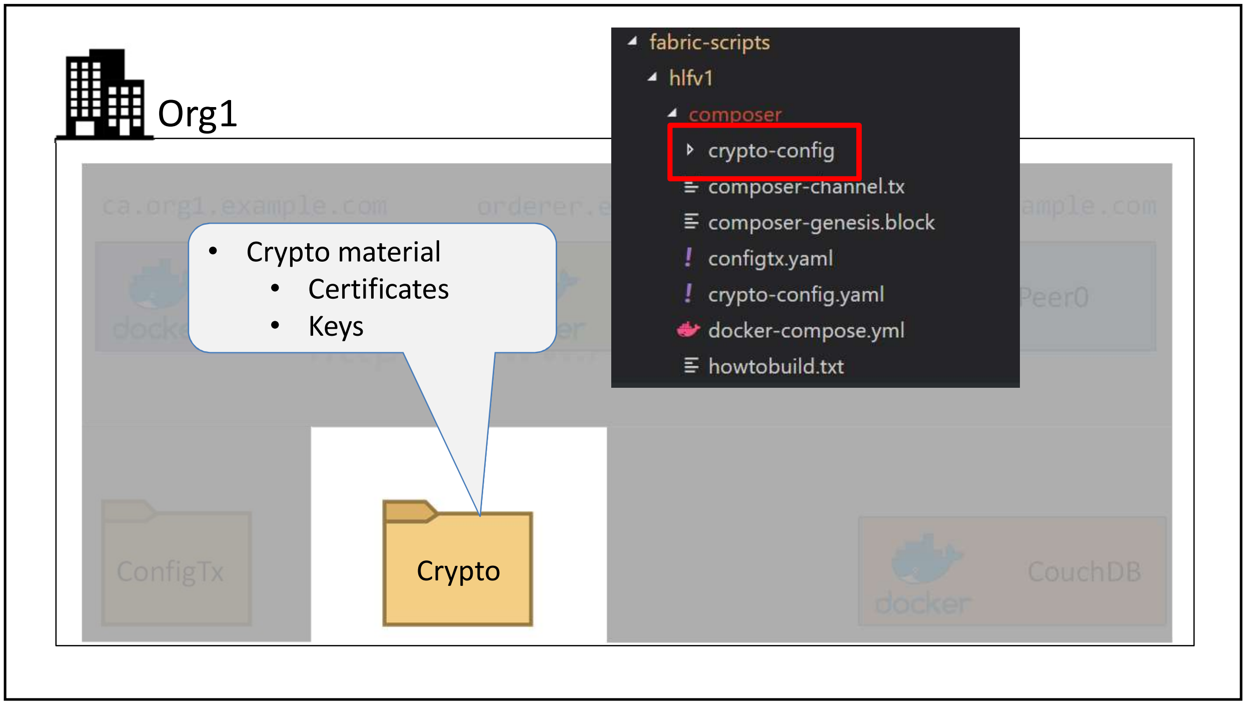






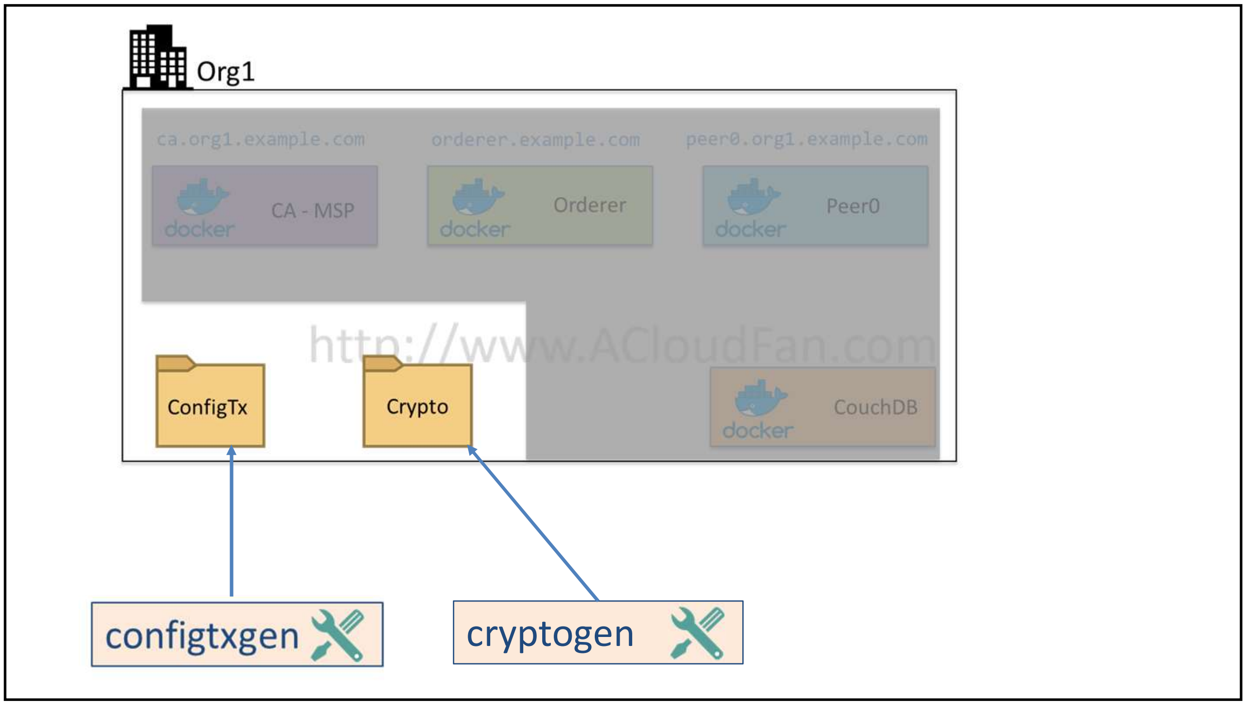
- Channel configuration
- Genesis block

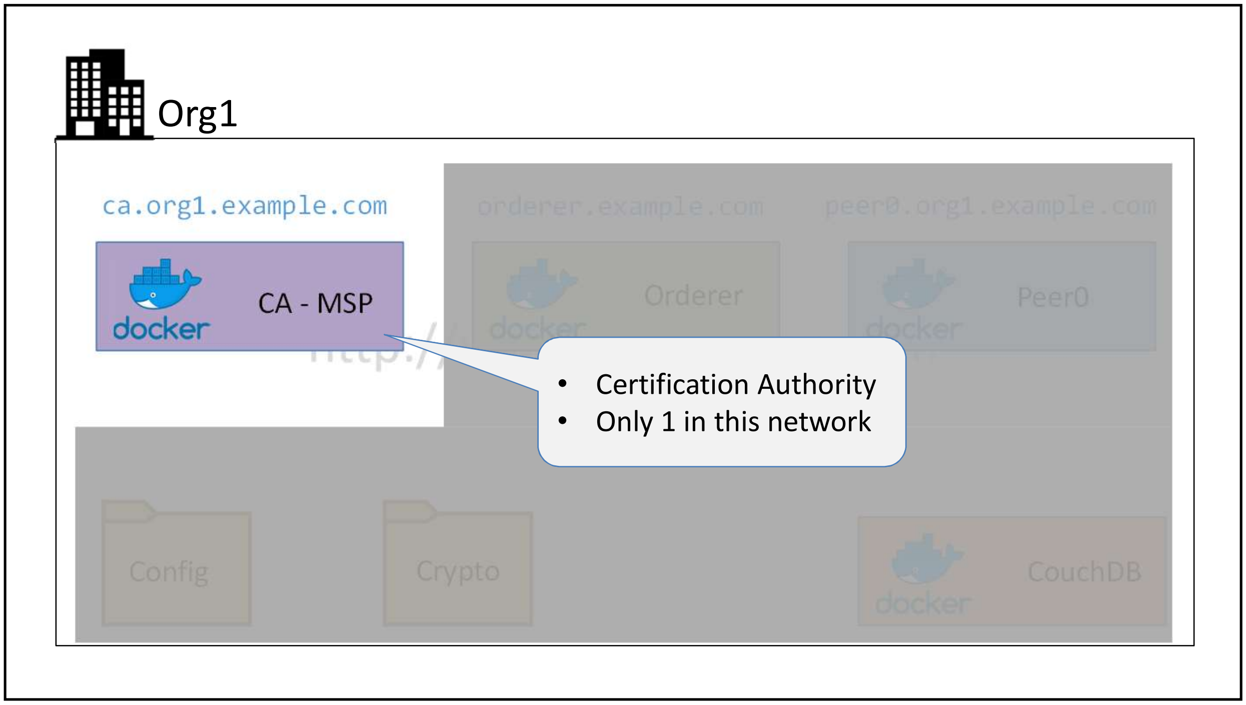
```
◢ fabric-scripts
  ◢ hlfv1
    ◢ composer
      ▷ crypto-config
        composer-channel.tx
        composer-genesis.block
      ! configtx.yaml
      ! crypto-config.yaml
        docker-compose.yml
        howtobuild.txt
```



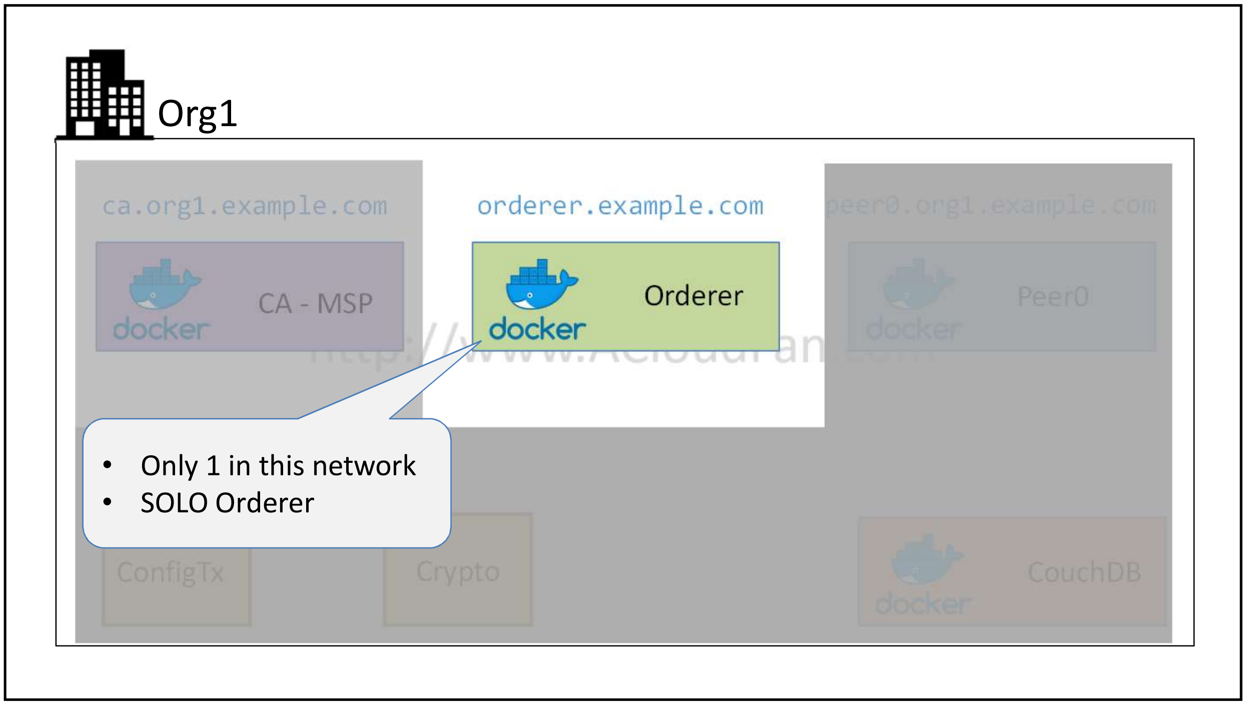
- Crypto material
  - Certificates
  - Keys

```
◢ fabric-scripts
  ◢ hlfv1
    ◢ composer
      ▷ crypto-config
        composer-channel.tx
        composer-genesis.block
      ! configtx.yaml
      ! crypto-config.yaml
        docker-compose.yml
        howtobuild.txt
```

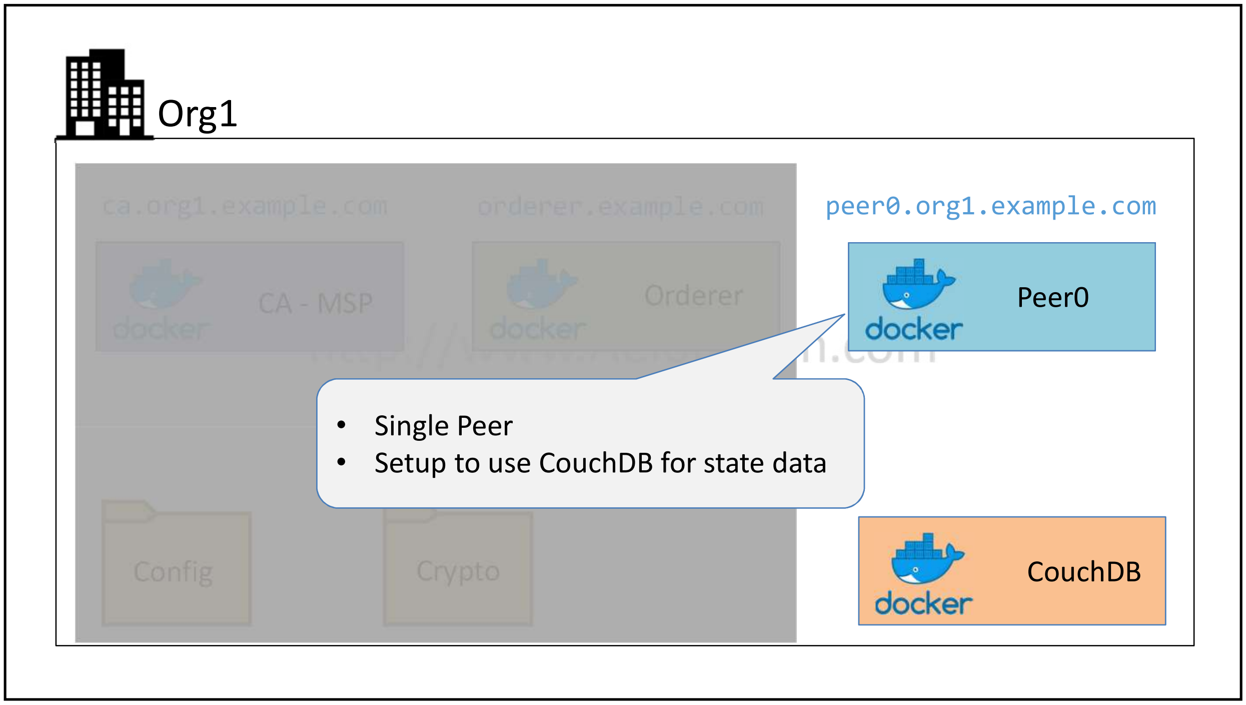




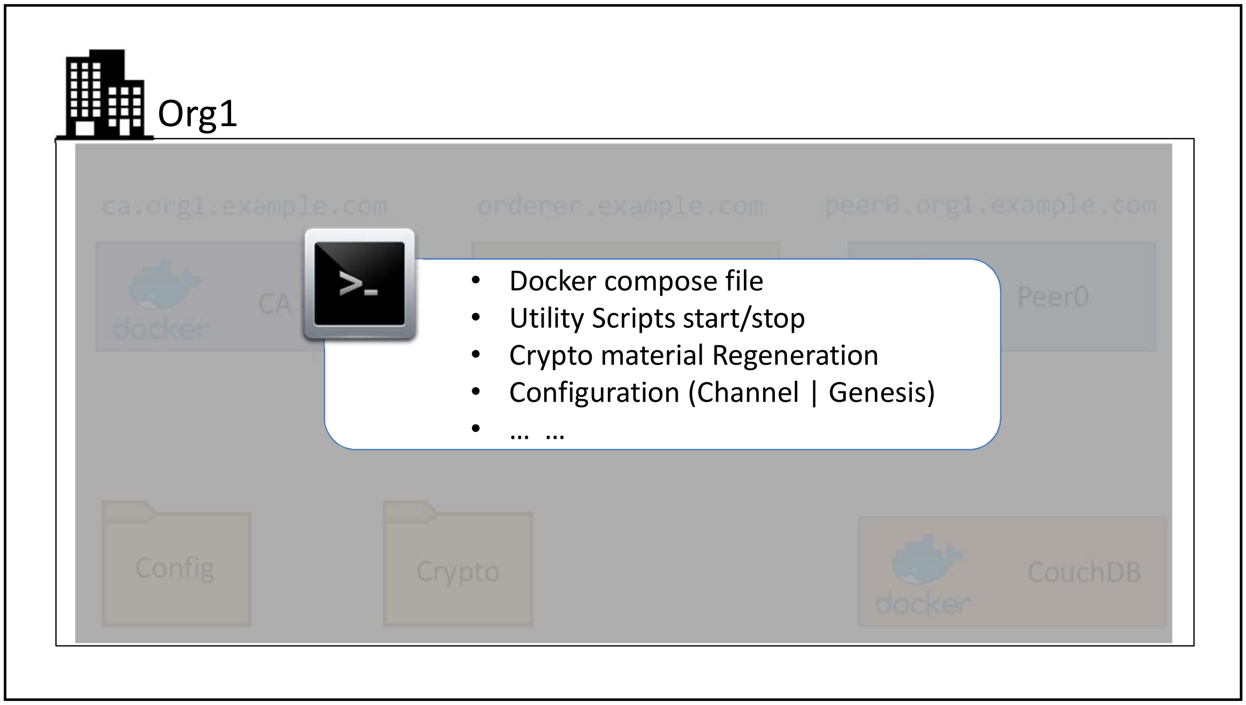
- Certification Authority
- Only 1 in this network



- Only 1 in this network
- SOLO Orderer

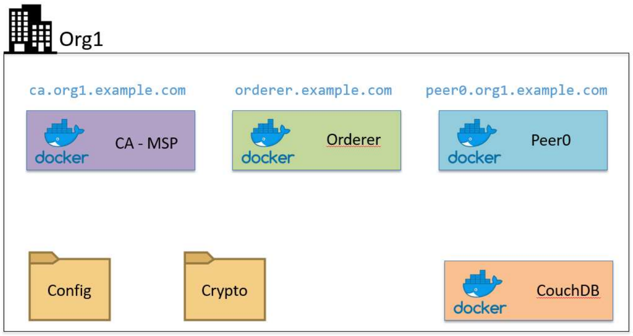


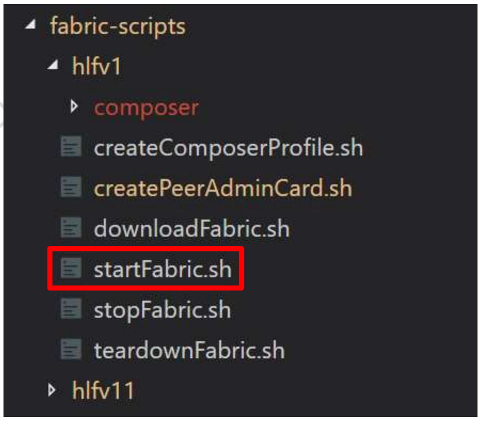
- Single Peer
- Setup to use CouchDB for state data



- Docker compose file
- Utility Scripts start/stop
- Crypto material Regeneration
- Configuration (Channel | Genesis)
- … …

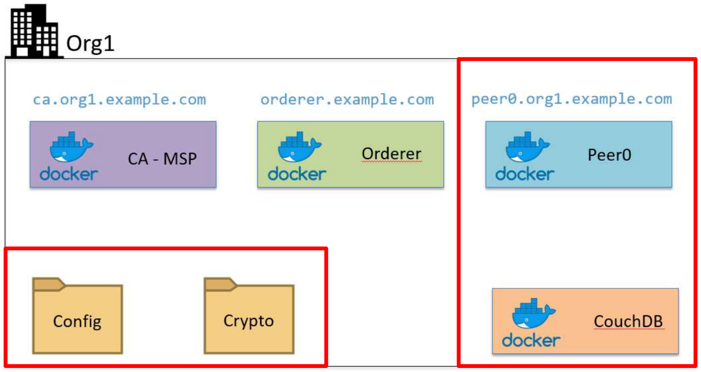
Summary





```
◢ fabric-scripts
  ◢ hlfv1
    ▷ composer
      createComposerProfile.sh
      createPeerAdminCard.sh
      downloadFabric.sh
      startFabric.sh
      stopFabric.sh
      teardownFabric.sh
  ▷ hlfv11
```

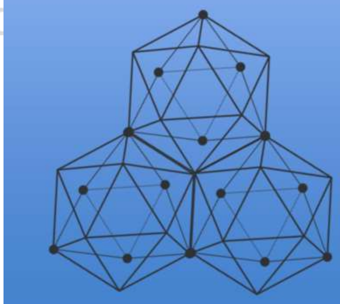
Walkthrough : Dev Setup



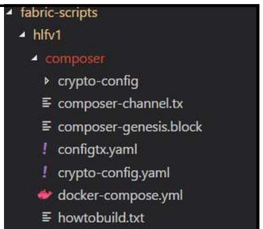
raj@acloudfan.com

@acloudfan

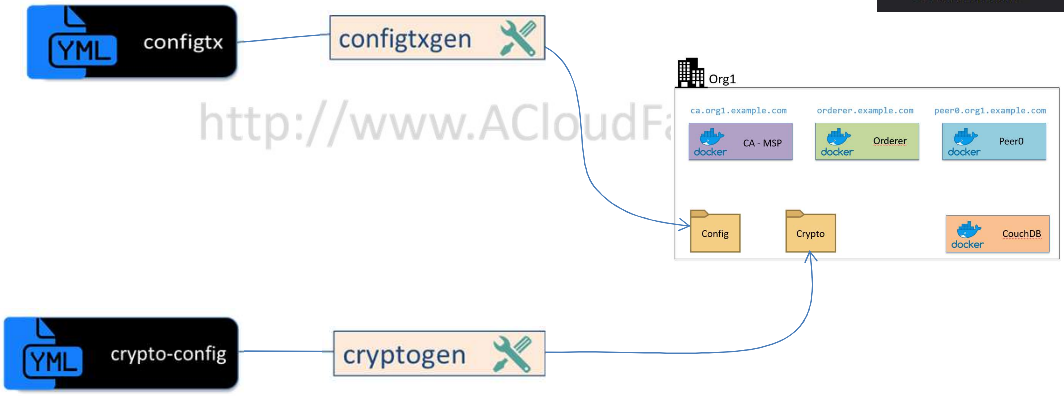
http://ACloudFan.com



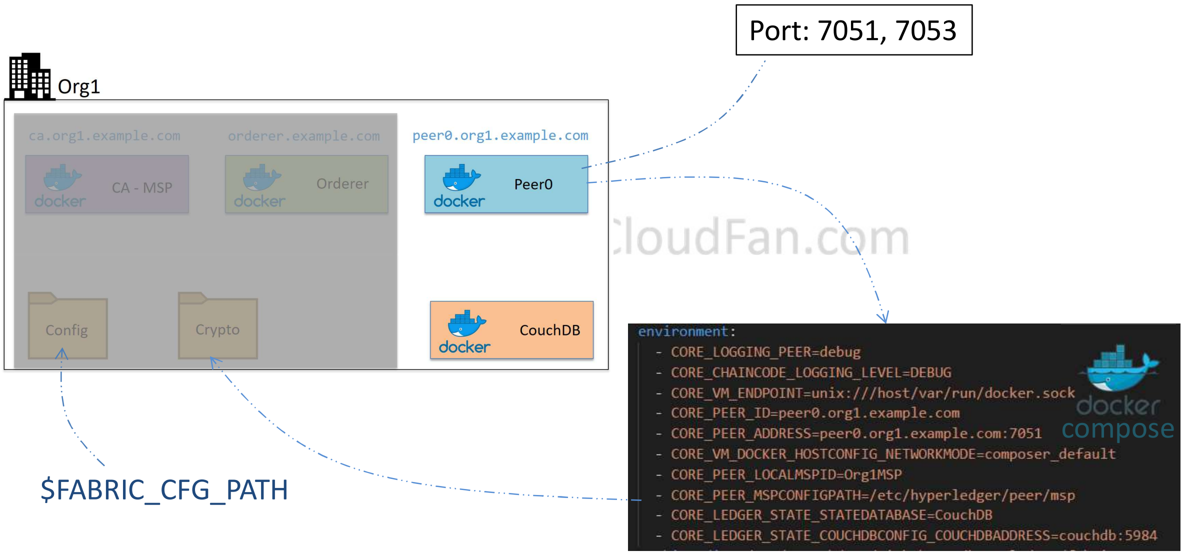
- Setup configured by way of 2 YAML files



```
◢ fabric-scripts
  ◢ hlfv1
    ◢ composer
      ▷ crypto-config
        composer-channel.tx
        composer-genesis.block
      ! configtx.yaml
      ! crypto-config.yaml
        docker-compose.yml
        howtobuild.txt
```

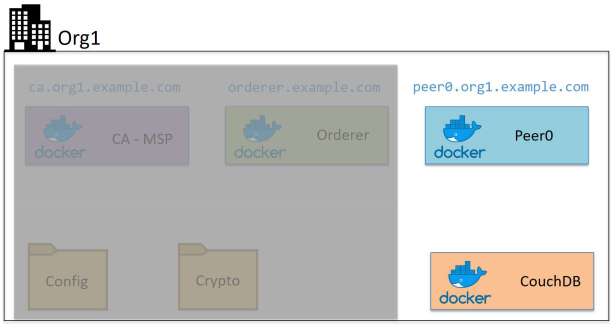


{Dev Setup}

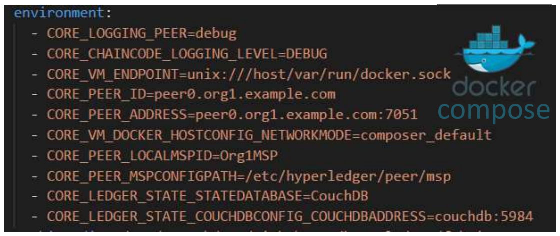


```
environment:
  - CORE_LOGGING_PEER=debug
  - CORE_CHAINCODE_LOGGING_LEVEL=DEBUG
  - CORE_VM_ENDPOINT=unix:///host/var/run/docker.sock
  - CORE_PEER_ID=peer0.org1.example.com
  - CORE_PEER_ADDRESS=peer0.org1.example.com:7051
  - CORE_VM_DOCKER_HOSTCONFIG_NETWORKMODE=composer_default
  - CORE_PEER_LOCALMSPID=Org1MSP
  - CORE_PEER_MSPCONFIGPATH=/etc/hyperledger/peer/msp
  - CORE_LEDGER_STATE_STATEDATABASE=CouchDB
  - CORE_LEDGER_STATE_COUCHDBCONFIG_COUCHDBADDRESS=couchdb:5984
```

{Dev Setup}



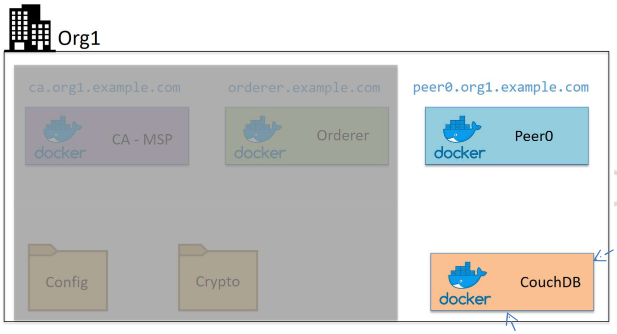
Peer binary is in the docker image

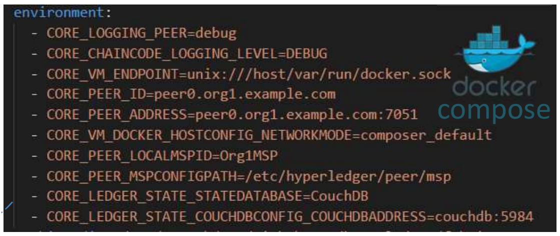


```
environment:
  - CORE_LOGGING_PEER=debug
  - CORE_CHAINCODE_LOGGING_LEVEL=DEBUG
  - CORE_VM_ENDPOINT=unix:///host/var/run/docker.sock
  - CORE_PEER_ID=peer0.org1.example.com
  - CORE_PEER_ADDRESS=peer0.org1.example.com:7051
  - CORE_VM_DOCKER_HOSTCONFIG_NETWORKMODE=composer_default
  - CORE_PEER_LOCALMSPID=Org1MSP
  - CORE_PEER_MSPCONFIGPATH=/etc/hyperledger/peer/msp
  - CORE_LEDGER_STATE_STATEDATABASE=CouchDB
  - CORE_LEDGER_STATE_COUCHDBCONFIG_COUCHDBADDRESS=couchdb:5984
```

{Dev Setup}

```
$ curl http://127.0.0.1:5984
$ curl http://127.0.0.1:5984/db/_all_dbs
```





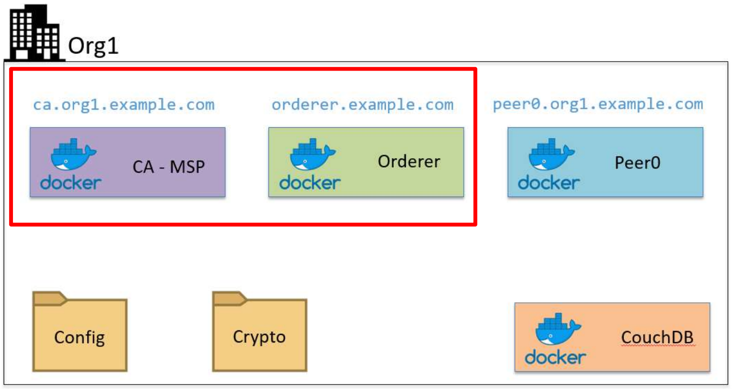
```
environment:
  - CORE_LOGGING_PEER=debug
  - CORE_CHAINCODE_LOGGING_LEVEL=DEBUG
  - CORE_VM_ENDPOINT=unix:///host/var/run/docker.sock
  - CORE_PEER_ID=peer0.org1.example.com
  - CORE_PEER_ADDRESS=peer0.org1.example.com:7051
  - CORE_VM_DOCKER_HOSTCONFIG_NETWORKMODE=composer_default
  - CORE_PEER_LOCALMSPID=Org1MSP
  - CORE_PEER_MSPCONFIGPATH=/etc/hyperledger/peer/msp
  - CORE_LEDGER_STATE_STATEDATABASE=CouchDB
  - CORE_LEDGER_STATE_COUCHDBCONFIG_COUCHDBADDRESS=couchdb:5984
```

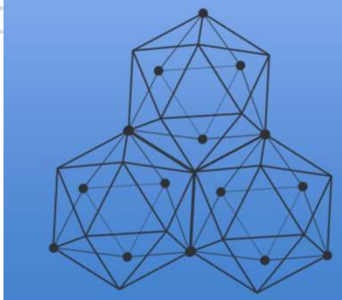
Walkthrough : Dev Setup

raj@acloudfan.com

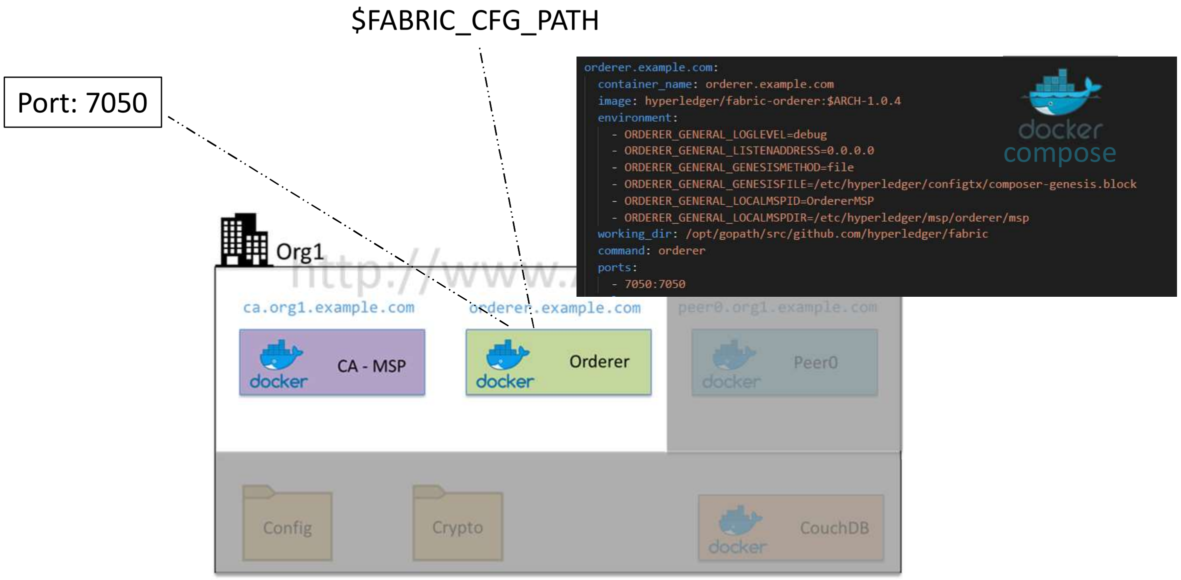
@acloudfan

http://ACloudFan.com





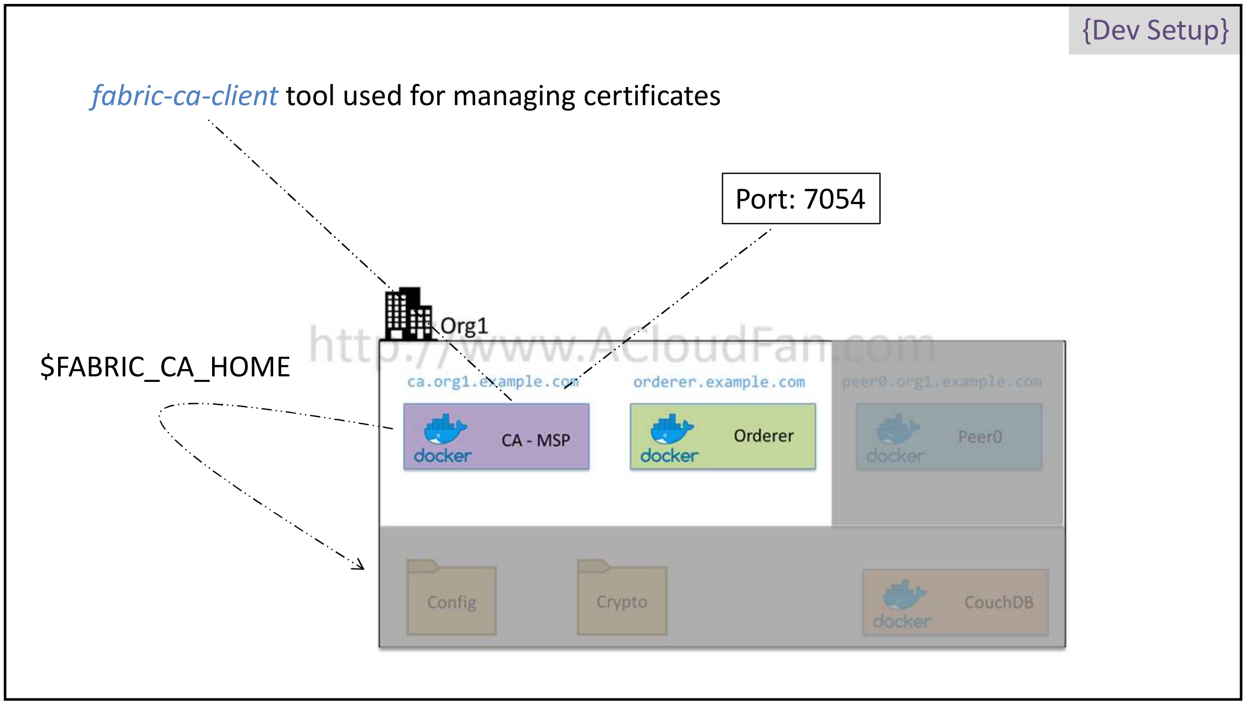
{Dev Setup}



$FABRIC_CFG_PATH

Port: 7050

```
orderer.example.com:
  container_name: orderer.example.com
  image: hyperledger/fabric-orderer:$ARCH-1.0.4
  environment:
    - ORDERER_GENERAL_LOGLEVEL=debug
    - ORDERER_GENERAL_LISTENADDRESS=0.0.0.0
    - ORDERER_GENERAL_GENESISMETHOD=file
    - ORDERER_GENERAL_GENESISFILE=/etc/hyperledger/configtx/composer-genesis.block
    - ORDERER_GENERAL_LOCALMSPID=OrdererMSP
    - ORDERER_GENERAL_LOCALMSPDIR=/etc/hyperledger/msp/orderer/msp
  working_dir: /opt/gopath/src/github.com/hyperledger/fabric
  command: orderer
  ports:
    - 7050:7050
```

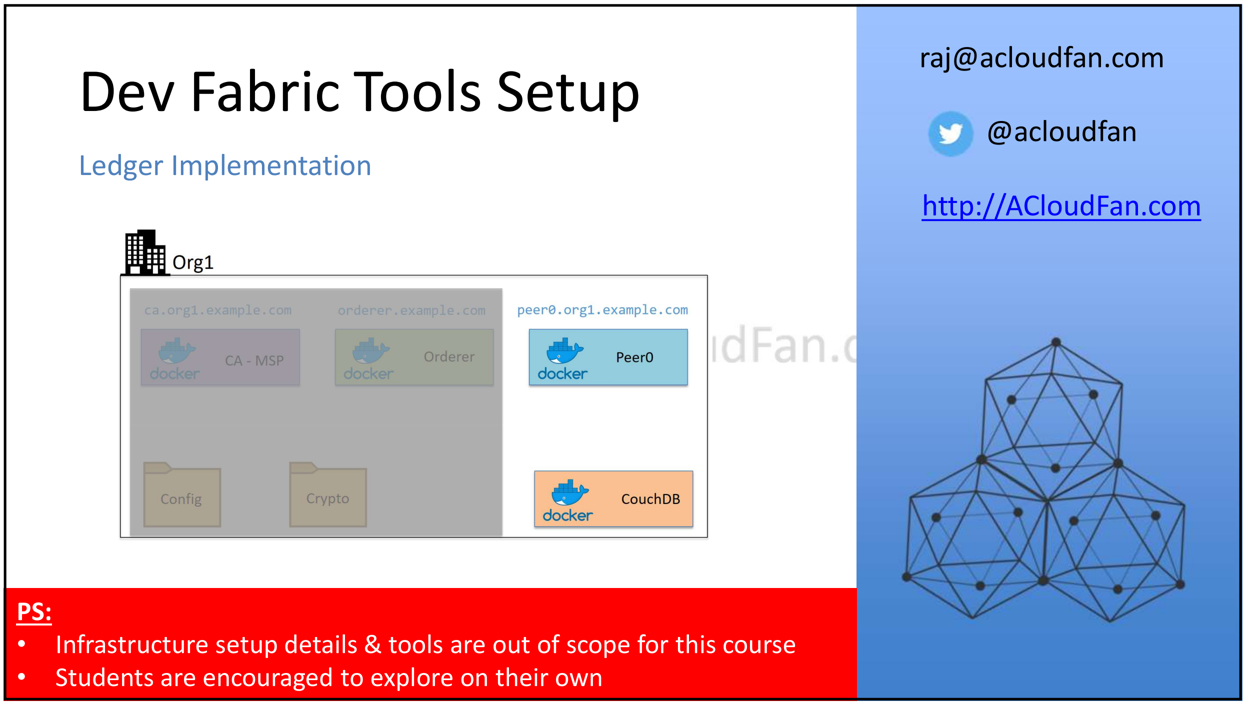


{Dev Setup}

fabric-ca-client tool used for managing certificates

Port: 7054

$FABRIC_CA_HOME



Dev Fabric Tools Setup

Ledger Implementation

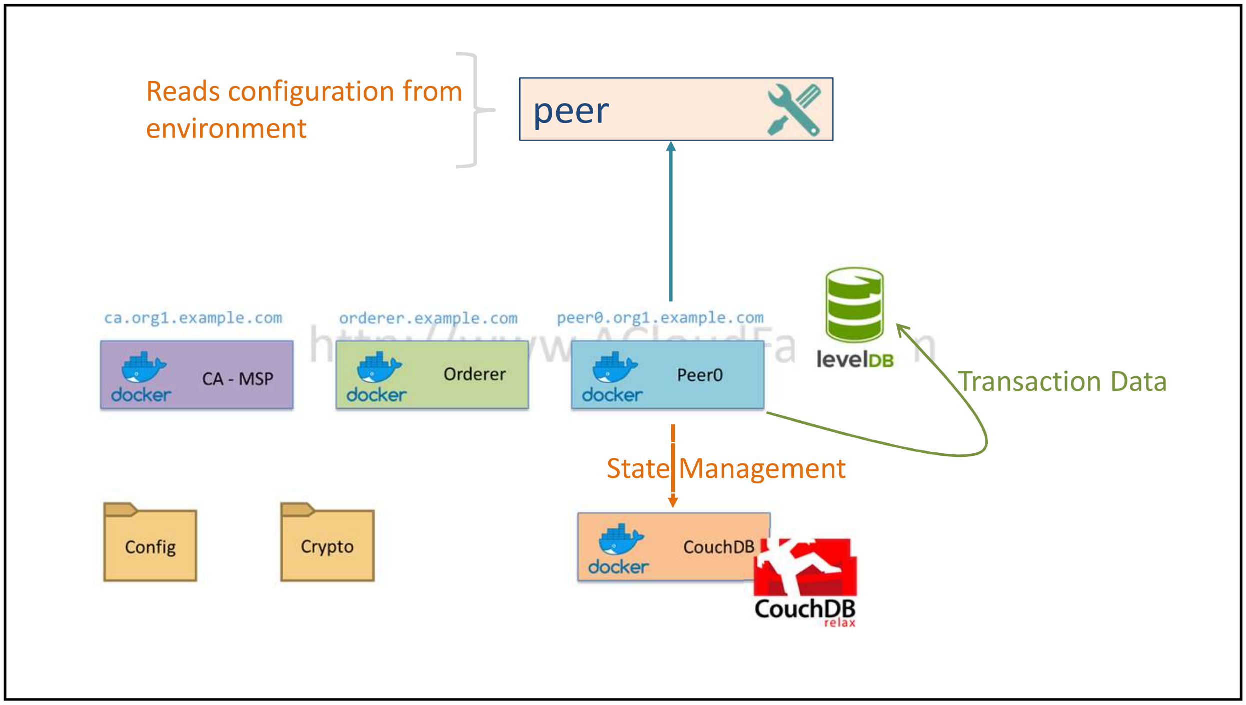
raj@acloudfan.com

@acloudfan

http://ACloudFan.com

PS:

- Infrastructure setup details & tools are out of scope for this course
- Students are encouraged to explore on their own



Reads configuration from environment

peer

ca.org1.example.com

CA - MSP

orderer.example.com

Orderer

peer0.org1.example.com

Peer0

levelDB

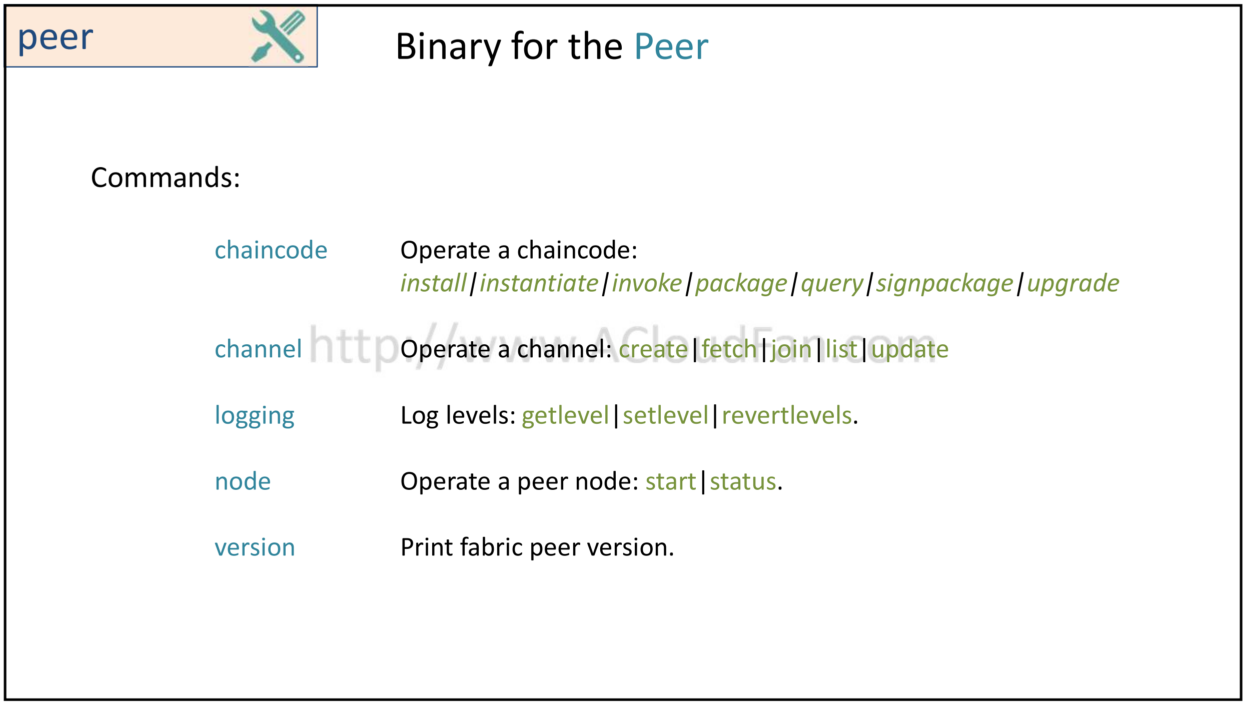
Transaction Data

State Management

Config

Crypto

CouchDB



peer

Binary for the Peer

Commands:

chaincode	Operate a chaincode: *install / instantiate / invoke / package / query / signpackage / upgrade*
channel	Operate a channel: create / fetch / join / list / update
logging	Log levels: getlevel / setlevel / revertlevels.
node	Operate a peer node: start / status.
version	Print fabric peer version.