



HashFlow

Hyperledger Fabric Deployment on Cloud-
AWS, IBM and Oracle

By Matt Zand- CEO of [HashFlow](#)



How to Deploy Hyperledger Fabric on AWS, IBM and Oracle

Webinar

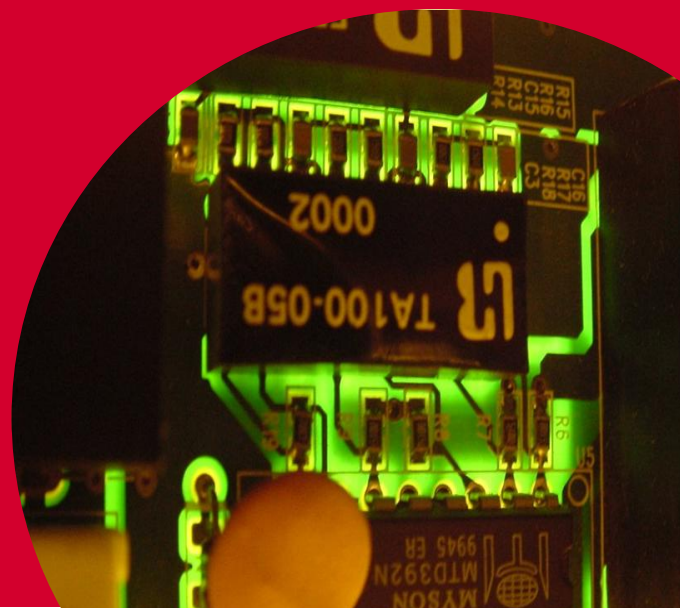
Recap

How much do you know about
Hyperledger?

- **Matt** is a serial entrepreneur and the founder of four tech startups: DC Web Makers, Coding Bootcamps, HashFlow and High School Technology Services.
- He is a leading author of Hands-on Smart Contract Development with Hyperledger Fabric book by O'Reilly Media.
- He has written more than 100 technical articles and tutorials on blockchain development for Hyperledger, Ethereum and Corda platforms.
- He has a master's degree in business management from the University of Maryland.
- Prior to blockchain development and consulting, he worked as senior web and mobile App developer and consultant, investor, business advisor for a few startup companies.

Outline

- Hyperledger Fabric Architecture
- HF on AWS
- HF on IBM
- HF on Oracle



1. Peer
2. Ordering service
3. Fabric CA (Certificate Authority)
4. Fabric ledger
5. Channel
6. Smart Contracts or chaincodes
7. Endorsement policy
8. Membership services provider (MSP)

Peer

- Peers can be created, started, stopped, reconfigured, and even deleted.
- They expose a set of APIs that enable administrators and applications to interact with the services that they provide.

Ordering Service

- Sometimes also referred as Orderer
- The mechanism by which applications and peers interact with each other to ensure that every peer's ledger is kept consistent with each other is mediated by special nodes called **orderers**
- Orderer Types
 - Solo (deprecated in v2.x)
 - Kafka (deprecated in v2.x)
 - Raft (recommended)

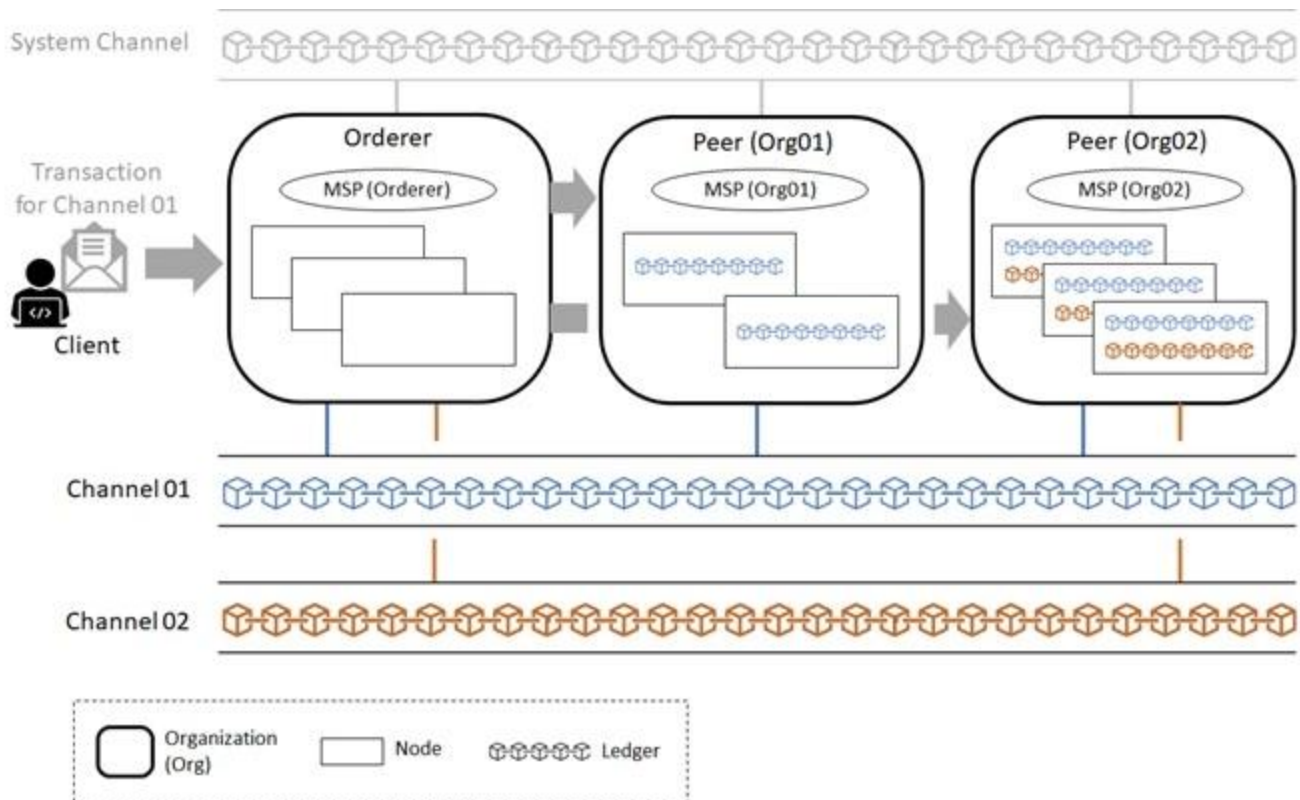
- Hyperledger Fabric is a consortium blockchain. The network of Hyperledger Fabric consists of each **Organizations (Org)** and CA. If needed, you can add new organization to consortium.
- The role of each organization is either of main 2 roles : **Orderer (Ordering Service)** or **Peer (Peer Service)**.
- Each participant company consists of one Peer. In each Peer organization, you can have multiple Peer nodes. Each peer node maintains the state and a copy of distributed ledger (chain).
- Orderer is a special organization in consortium, which coordinates all transactions in the network. When transactions are published, the orderer service validates, orders, and bundles transactions into blocks. (This process is sometimes called “ordering”.) And it broadcasts to all Peers in a channel.

Orderer’s organization also has multiple nodes, on which all blocks are validated by a consensus protocol.

- The entire network can have multiple ledgers and each ledger is maintained as **Channel**. Channel is like a virtual network, and physically it’s a ledger itself, which is distributed and decentralized in Peer’s nodes.

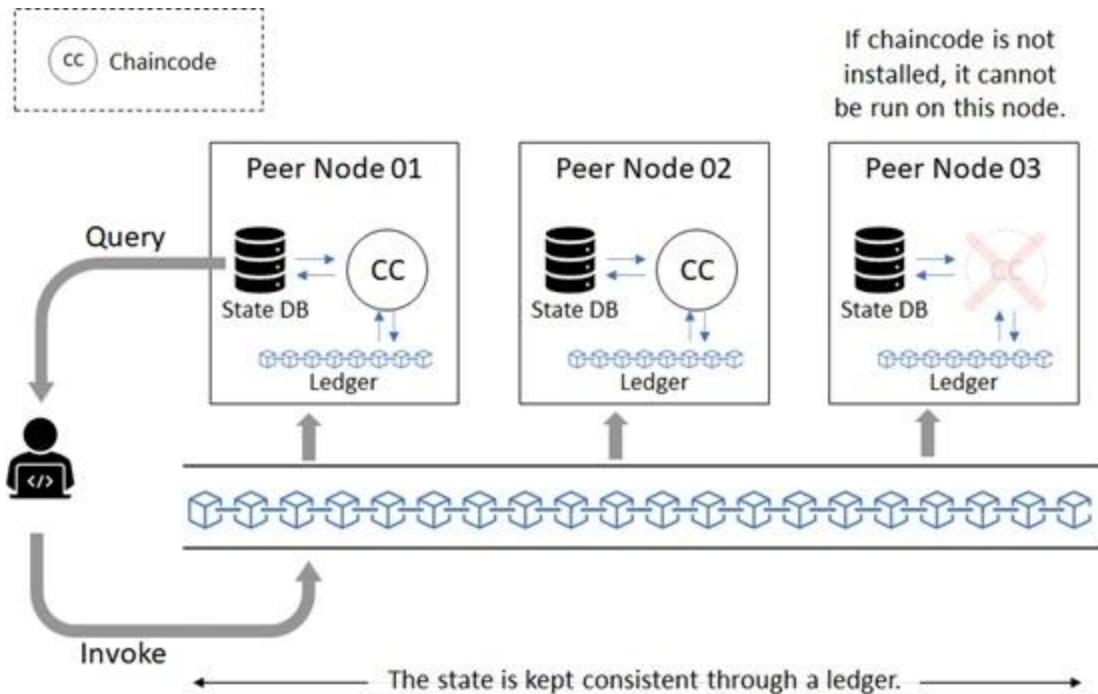
- Each organization (Peer) can participate multiple channels. For instance, channel01 has Org01, Org02, and Org03 as participant parties, whereas channel02 has only Org02 and Org03.
- The participants, endorsements, and validation policies are maintained in each channel.
- The network (consortium) has a special channel, called system channel. This channel is for administrating the entire network.
- As you know, a blockchain needs a set of certificates. In Hyperledger Fabric, these certificates (including Root CA certificate, keystore, etc) are packaged as **MSP (Membership Service Provider)**.
- Each Peer organization has corresponding their own MSP. When some new organization joins in some channel, this organization first starts to enroll MSP in genesis block or channel. (All organizations participating in a network should be enrolled in consortium on system channel.)
- Even when a client (SDK) connects to this organization, this client requires a connection profile, which is a part of MSP.

HF Architecture



Chaincodes

- When you want to execute chaincode in some node, the code should be installed in this node. The result state is written in **State Database (State DB)**, and its transaction (i.e, journal) is logged on a ledger.
- The state database is generic key-value database (you can choose LevelDB or CouchDB for state database) and might be easily modified, then you might think it's not immutable. However, the transaction's logs are immutably maintained in a decentralized ledger, then the state database can be regenerated from the chain at any time, and automatically get recovered (or generated if needed) upon peer startup. Therefore the state is kept consistent through a ledger.



Fabric on AWS

I. Overview of Amazon Blockchain Services

II. Setup Amazon Managed Blockchain

- Signup for AWS
- Create Virtual Private Cloud (VPC) and subnet
- Create security group
- Create EC2 instance (as a Bastion host)
- Elastic IP (or static IP)
- Create AWS (Identity and Access Management) IAM policy
- Create AWS IAM Role

III. Setup Hyperledger Fabric Network

- Create the Network and First Member
- Create VPC endpoint

Fabric on AWS

IV. Setup the Hyperledger Fabric Client

- Install Packages
- Set Up the Hyperledger Fabric CA Client
- Clone the Samples Repository
- Enroll an Administrative User
- Create a Peer Node in Your Membership
- Configure and Run Docker Compose to Start the Hyperledger Fabric CLI
- Create configtx for Hyperledger Fabric Channel Creation
- Set Environment Variables for the Orderer
- Create the Channel
- Join Your Peer Node to the Channel
- Install Chaincode
- Instantiate Chaincode
- Query the Chaincode
- Invoke the Chaincode

Fabric on AWS

AWS (Amazon Web Services) provides the following three ways to create blockchain services:

Amazon Quantum Ledger Database (QLDB)

Amazon QLDB is a fully managed No-SQL(Semi-SQL & Semi-NoSQL) immutable and transparent ledger database powered by blockchain. Append-only data with cryptographically verifiable transaction history and audit log is owned by a central trusted authority. AWS QLDB automatically scales when applications need more capacity for read and write limits.

AWS Blockchain Templates

AWS Blockchain Templates enable you a fast and easy way to create and deploy blockchain project on the network and testing application on AWS platform. With building users pre-set blockchain frameworks through AWS CloudFormation stacks, it helps developers focus on building blockchain applications and spend less time to set up a blockchain network. It currently provides two types of blockchain template, Ethereum and Hyperledger Fabric.

Fabric on AWS

AWS (Amazon Web Services) provides the following three ways to create blockchain services:

Amazon Managed Blockchain

Amazon Managed Blockchain is a fully managed service, it provides a production level blockchain network. It can easily be scaled to support thousands of applications running millions of transactions. You can use AWS Management Console, the AWS Command Line Interface (CLI), or the Managed Blockchain Software Development Kit (SDK) to set up a managed blockchain application quickly.

Fabric on IBM

As one of the major contributors to the Hyperledger Fabric project, IBM offers a complete and intuitive platform, IBM Blockchain Platform, for building a Fabric network and deploying Fabric applications.

IBM Cloud allows users to fully manage their deployments, certificates, and private keys. It provides a simplified UI that allows developers to easily deploy Fabric components into a Kubernetes/Red Hat OpenShift cluster.

Fabric on IBM

Understand the IBM Platform Features

In a nutshell, IBM Blockchain Platform features can be divided into three areas: **development**, **operation**, and **scalability**.

Development

While developing Fabric applications, IBM offers a set of developer tools that simplifies the job of development and deployment. In particular, you can do the following:

- Use Ansible Playbooks or Red Hat Marketplace to deploy networks faster
- Go from proof of concept to production in a single environment by increasing Kubernetes resources
- Leverage IBM Blockchain Platform developer tools to easily code

Fabric on IBM

Operation

While running a Fabric application on IBM Cloud, you can take advantage of these great features:

- Ability to deploy peers to multiple channels on multiple clouds, or to permit other members to join your consortium.
- The Unified Codebase feature allows you to run your components on any environment backed by IBM Cloud and third-party, public clouds.
- The Dynamic Signature Collection feature facilitates collaborative governance over channel configurations.
- Removal of Docker-in-Docker for chaincodes makes chaincode pods run more securely, without peers requesting privileged access.
- Ability to easily upgrade nodes running on older Fabric versions to the latest version of Fabric. Upon upgrade, the capabilities of your channels and network will also be increased accordingly.
- Integration of Logging and Monitoring tools leveraging IBM Cloud services like IBM Log Analysis; IBM Cloud Monitoring is helpful for managing the blockchain network

Fabric on IBM

Scalability

As a prominent performance factor, scalability plays a vital role in the growth of Fabric applications. IBM Cloud gives you the following capabilities for scalability:

- Flexibility on the amount of compute power being utilized in your network by setting the amount of CPU, memory, and storage you wish to provision in your Kubernetes cluster
- Ability to adjust the amount of resources being utilized in your network by easily scaling up and down the resources in your Kubernetes cluster
- Ability to replicate your Kubernetes deployment among multiple regions or zones to ensure high availability of your components as well as disaster recovery

Fabric on IBM

Further, using the IBM Blockchain Platform console, you can easily create and manage blockchain applications. Specifically, once a blockchain service instance is provisioned on IBM Cloud, you can link it to your Kubernetes cluster, after which you can create and manage your blockchain components like CA, peer, and ordering service images through the console.

However, before using the console, you should know that you will be responsible for the following:

- Monitoring the health, security, and logging of Kubernetes clusters.
- Managing and securing your certificates and private keys. IBM does not save your certificates in the Kubernetes cluster

Fabric on IBM

1. Create an IBM Cloud Account
2. Decide the Structure of the Blockchain Network
As an example, let's say our Fabric network includes the following: Two CAs, Two MSPs, One peer, One Orderer, One channel that includes one peer org and one Orderer org
3. Create and Join the Fabric Network
4. Build the Blockchain Network
 1. Add a certificate authority
 2. Associate the admin identity
 3. Register users
5. Create the peer organization's MSP definition
6. Create the peer node
7. Create the ordering service
8. Add the organization as a consortium member
9. Create & Join a channel
10. Deploy the smart contract
11. Create the application

Fabric on Oracle

- Oracle Blockchain Platform Overview
- Oracle Blockchain Platform Architecture
- Solution Design
 - Explore the use case
 - Engage with the solution
 - Experimenting with Oracle Blockchain Platform
 - Oracle cloud Infrastructure Configuration
 - Create OBP Instances
 - Create Blockchain Network
 - Create Transaction Infrastructure
 - Experience the Solution
 - Adding the smartness

Fabric on Oracle

Oracle blockchain product offering include following products:

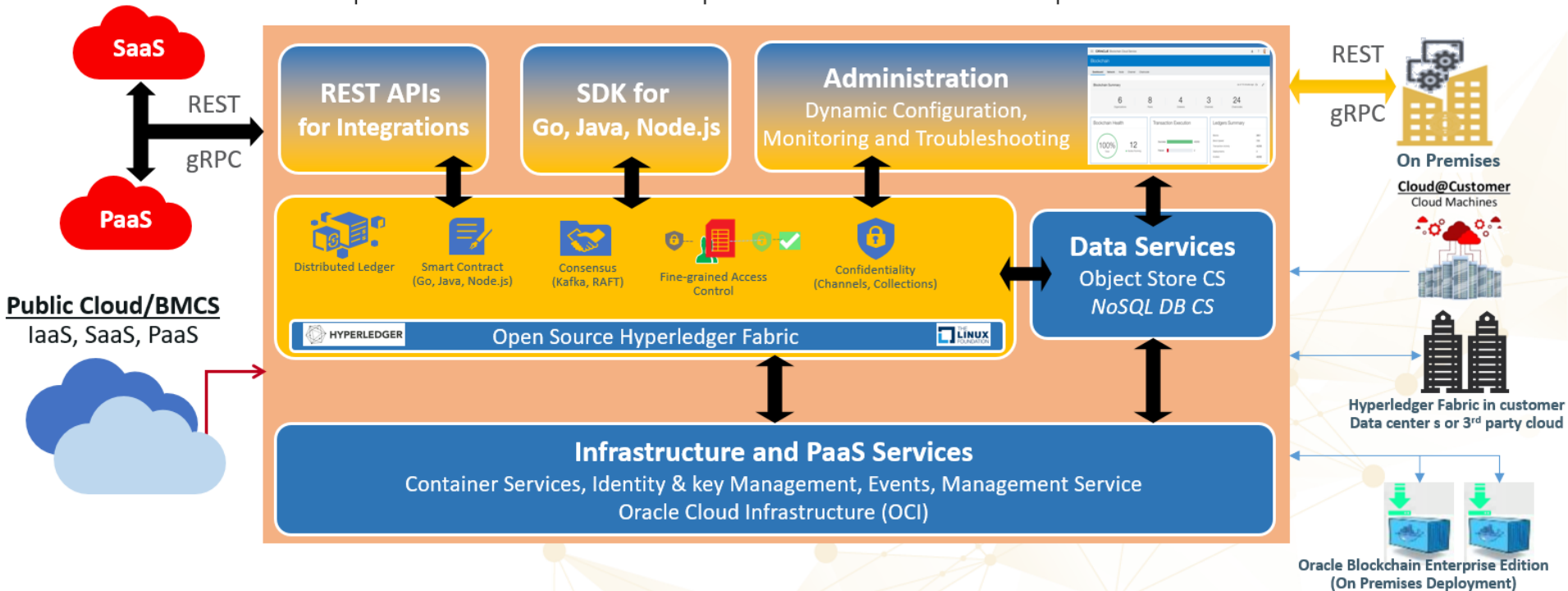
Oracle Blockchain Platform Cloud Service – It is a cloud offering which includes preassembled Platform as a Service (PaaS) with a template based provisioning. It leverages OCI which takes care of dependencies like containers, virtual machines, identity management, block and object storage.

Oracle Blockchain Platform Enterprise edition – It's for those customers who need greater data privacy or cannot run on cloud. Oracle blockchain platform enterprise edition, offers preassembled, on-premises blockchain, which can be deployed on your data center. It has feature parity with cloud equivalent blockchain cloud and offers the same APIs and portability of applications.

Prebuilt solution - It offers pre-integrated, quickly deployable blockchain solutions. These solutions can seamlessly connect with SaaS applications like SCM etc. For example, intelligence track and trace is a pre-integrated blockchain solution to achieve end-to-end supply chain visibility.

Tamper-resistant blockchain tables in Oracle Database – It's a solution for an enterprise that needs immutable storage for use cases like tamper resistant transaction logs, audit trails, compliance data, financial ledger or legal hold data etc. It is a tamper resistant option in Oracle database that only allows insert operations. Each row contains the previous row's hash thus chaining the rows. Non-repudiation is guaranteed as user's sign the content of the rows with PKI based signature.

Pre-assembled | Managed service | Enterprise-ready | Public or private cloud
Open | Plug & Play Integration | Production-ready | Automated Dev-Ops





Explore



Engage



Experiment



Experience

OCI Configuration

Plan blockchain edition

Sign-in or Sign-up

Create Compartment

Create Instance

Create OBP Instance

Create Blockchain Network

Export Participant(s) organization certificate

Import Participant(s) certificate to founder Organization

Export Orderer setting from founder

Import Orderer setting into participant(s)

Create Transaction Infrastructure

Create Channel at founder

Add Participant's peer node to channel

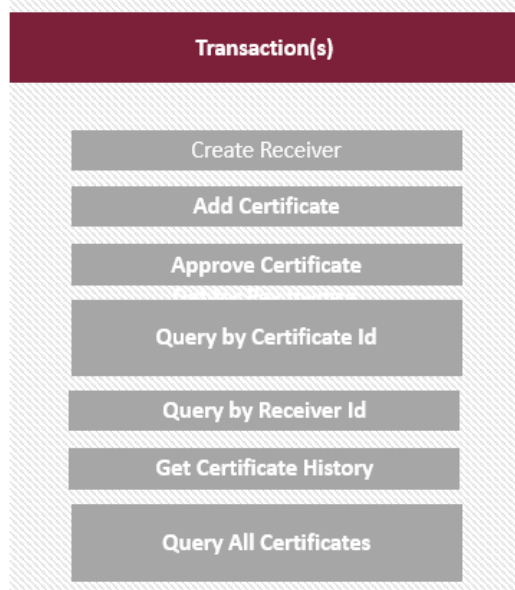
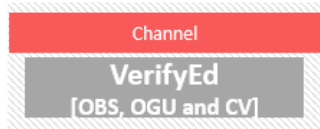
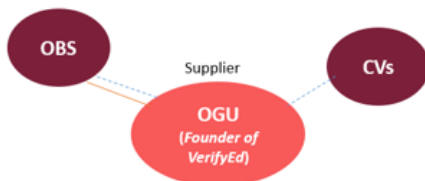
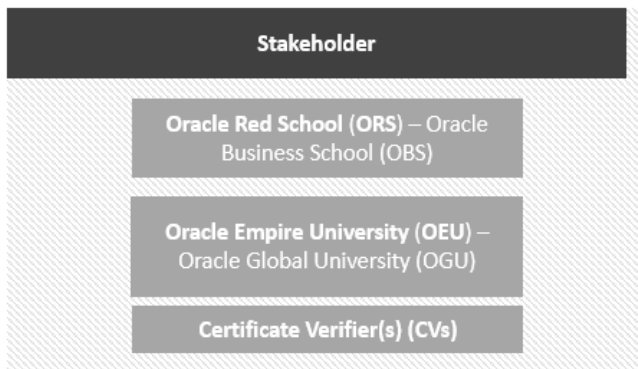
Adding Smartness

Develop Chaincode

Deploy Chaincode

Instantiate Chaincode

Test Chaincode



Access Control List

Participant ID	Read-Only	Read & Write
OBS		<input checked="" type="checkbox"/>
OGU		<input checked="" type="checkbox"/>
CVs	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

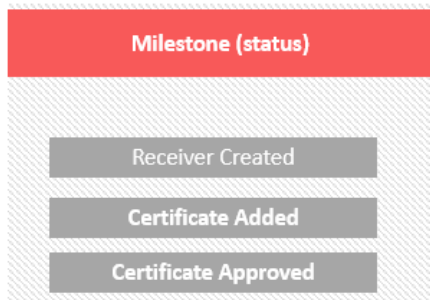
Asset

Learner

- Receiver_id
- Receiver_name
- Receiver_org

Certificate

- Cert_id
- Cert_no
- Cert_name
- Cert_receiver
- Cert_receiver_id
- Cert_issuer
- Cert_industry
- Cert_create_time
- Cert_update_time
- Cert_remark
- Cert_status



- System admin's main tasks are:
 - Application Lifecycle Management
 - Install and Configure Network
 - Diagnostics and Troubleshooting including logs
 - Membership Service Provider
 - Network Maintenance and Operations

- System admin technical requirements:

From technical standpoint, to become a professional HF system admin, one should acquire a basic knowledge of the Linux command line, bash, Docker and containers (like Kubernetes), NoSQL, CouchDB, and blockchain and distributed ledgers.

- Certification exam for system admins:

Certified Hyperledger Fabric Administrator
(CHFA) certification is highly recommended.

- The role of Fabric Developer is to:

In comparison with system administration, HF developers' primary task is to work on one major component of HF which is smart contract or chaincode. In short, developers are required to design, develop, test and deploy HF chaincodes on the peer.

- Fabric developer's main tasks are:
 - Define transaction functions
 - Execute simple queries
 - Create complex queries
 - Define assets using key value pairs
 - Identify potentially private data
 - Incorporate private data collection
 - Submit, evaluate, and query transaction by invoking the smart contracts

- Technical requirements for Fabric developers:

HF developers must have a solid foundation or background in one of the following programming languages: JavaScript, Java, Go, or Python. The most famous JS framework that is currently used by Fabric developers is Node.JS.

- Certification exam for Fabric developers:

Certified Hyperledger Fabric Developer certification is highly recommended.

- How HashFlow Can Help

At HashFlow, we are helping enterprise clients in all stages of Hyperledger Fabric product development from PoC to production on popular cloud providers- AWS, IBM and Oracle.

- How HashFlow Can Help

For further assistance, you can contact me directly on LI
<https://www.linkedin.com/in/matt-zand-64047871>

Or contact us via our website: <https://www.hashflow.us/>

Recap

What we learned in this webinar?

Resources- Hyperledger Training

[Hyperledger Fabric for System Admins](#)

[Hyperledger Fabric for Developers](#)

[Intro to Hyperledger Sawtooth for System Admins](#)

Resources- AWS Training

AWS certifications is highly recommended:

- AWS Cloud Practitioner certification
- AWS Solutions Architect Associate certification
- AWS Developer Associate certification
- AWS SysOps Administrator certification
- AWS Security Specialist certification

Resources- Book

For an in-depth coverage of topics discussed in this webinar, reading our below book is highly recommended.

Hands-on Smart Contract Development with Hyperledger Fabric:
<https://www.amazon.com/gp/product/1492086126/>

Q/A



HashFlow

Thank You

Hashflow.us

