

IBM Blockchain

Three introductory labs

Lennart Frantzell, Syed Zaidi
October 30, 2016



Blockchain agenda

1. Double-entry bookkeeping and the business background to the Blockchain
2. The Hyperledger Project, Blockchain architecture, concepts and terminology
3. Three Blockchain hands-on labs
4. Where do we go from here?

Blockchain intro

<https://www.ibm.com/developerworks/cloud/library/cl-blockchain-basics-intro-bluemix-trs/index.html>

<https://www.ibm.com/developerworks/cloud/library/cl-ibm-blockchain-101-quick-start-guide-for-developers-bluemix-trs/index.html> **IBM Blockchain 101: Quick-start guide for developers**

Section 1:

Double-entry bookkeeping and the business background to the Blockchain

Double-entry bookkeeping and the role of business ledgers

LEDGER

ACCOUNT TYPE		CASH			
TRANSACTION DATE	TRANSACTION DETAIL	REFERENCE	DEBIT	CREDIT	BALANCE
1/1/16	Expenses for Jan	Ref#1	\$100.00		\$100.00
2/1/16	Tax withheld	Ref#2		\$110.00	(\$10.00)

Luca Bartolomeo de Pacioli



Portrait of Luca Pacioli, traditionally attributed to *Jacopo de' Barbari*, 1495 (attribution controversial).^[1]

- Business transactions typically involve various participants like buyers, sellers, and intermediaries (such as banks, auditors, or notaries) whose business agreements and contracts are recorded in **business ledgers**.
- A business typically uses multiple ledgers to keep track of asset ownership and asset transfers between participants in its various lines of businesses.
- Ledgers are the systems of record (SORs) for a business's economic activities and interests.

Problems with current business ledgers

Current business ledgers in use today are deficient in many ways:

- They are inefficient, costly, non-transparent, and subject to fraud and misuse.
- These problems stem from centralized, trust-based, third-party systems,
- such as financial institutions, clearing houses, and other mediators of existing institutional arrangements.
- Centralized, trust-based ledger systems lead to bottlenecks and slowdowns of
- transaction settlements.
- Lack of transparency, as well as susceptibility to corruption and fraud, lead to disputes.
- Having to resolve disputes and possibly reverse transactions or provide Insurance for transactions is costly.

.....

Lets look at some of the problems that Blockchain can solve

<https://www.youtube.com/watch?v=lgNfoQQ5Reg> IBM Blockchain Car Lease Demo

<https://www.youtube.com/watch?v=F0P7NM7d-ps> Blockchain in IBM Global Financing

<https://www.youtube.com/watch?v=TZjs-BYiE5I&feature=youtu.be> Blockchain on IBM z Systems for Finance

<https://www.youtube.com/watch?v=r0LsnzAe1Yg> International Trade Solution on Blockchain

<https://www.youtube.com/watch?v=B2eYh2Y7rJw&feature=youtu.be> Trading Commerical Paper via blockchain with IBM Bluemix

Blockchain intro

Overview

Our interactive demo tracks the shipment of a package as it moves along the supply chain, passing through multiple carriers. It shows how an IoT-enabled package transmits required status information along the way and how each transaction is tracked and shared among all partners in a blockchain.

Play




<https://www.ibm.com/internet-of-things/iot-news/announcements/private-blockchain/>


Section 2:

The Hyperledger project, concepts and technologies

The Hyperledger project



Hyperledger Project 

 <https://www.hyperledger.org>

<https://github.com/hyperledger>

The Hyperledger project

<https://www.hyperledger.org/>



The image shows a screenshot of the Hyperledger Project's Twitter profile and a recent tweet. The profile header includes the Hyperledger logo (a geometric network of nodes and lines) and the text "HYPERLEDGER PROJECT". Below this is a description: "The Hyperledger Project is an open source, industry-wide project that brings together future and existing open source blockchain technology. Governed by The Linux Foundation, the project is building a general-purpose blockchain framework that can be used across industry sectors, from financial services to retail to manufacturing and more." and a link to "hyperledger.org". The profile statistics show 1,511 tweets, 3,133 following, 7,262 followers, 273 likes, and 4 lists. A "Follow" button is visible. The tweet section shows a tweet from "Hyperledger Project @Hyperledger" with 55m retweets, titled "Australia reveals plans for Blockchain in public services". The tweet content includes an icon of a document and the text: "Exclusive: Australia reveals plans for Blockchain in... Foresight unit discusses potential in police, food security, and government communications. govinsider.asia".

HYPERLEDGER PROJECT

The Hyperledger Project is an open source, industry-wide project that brings together future and existing open source blockchain technology. Governed by The Linux Foundation, the project is building a general-purpose blockchain framework that can be used across industry sectors, from financial services to retail to manufacturing and more.

Learn more at hyperledger.org.

TWEETS 1,511 FOLLOWING 3,133 FOLLOWERS 7,262 LIKES 273 LISTS 4

[Follow](#)

Hyperledger Project
@Hyperledger

An industry-wide open source project to advance blockchain technology and governed by The Linux Foundation.

San Francisco, CA
hyperledger.org
Joined May 2014

Tweets Tweets & replies Media

Hyperledger Project @Hyperledger · 55m
Australia reveals plans for Blockchain in public services

Exclusive: Australia reveals plans for Blockchain in...
Foresight unit discusses potential in police, food security, and government communications.
govinsider.asia

<https://twitter.com/Hyperledger>

<https://www.youtube.com/watch?v=EKa5Gh9whgU> Building a blockchain for business with the Hyperledger Project

The Hyperledger project

- A cross-industry collaborative effort, started in December 2015 by the Linux Foundation to support blockchain-based distributed ledgers
- It is focused on ledgers designed to support global business transactions, including major technological, financial, and supply chain companies, with the goal of improving many aspects of performance and reliability.
- The project aims to bring together a number of independent efforts to develop open protocols and standards, by providing a modular framework that supports different components for different uses. This would include a variety of blockchains with their own consensus and storage models, and services for identity, access control, and contracts.

<https://twitter.com/Hyperledge>

<https://www.youtube.com/watch?v=EKa5Gh9whgU> Building a blockchain for business with the Hyperledger Project

The Hyperledger project

The Hyperledger Project defined a protocol specification known as **Open Blockchain Protocol Specification** to create a blockchain fabric for use in a variety of industry use-cases involving B2B and B2C transactions. The primary goals of this effort are:

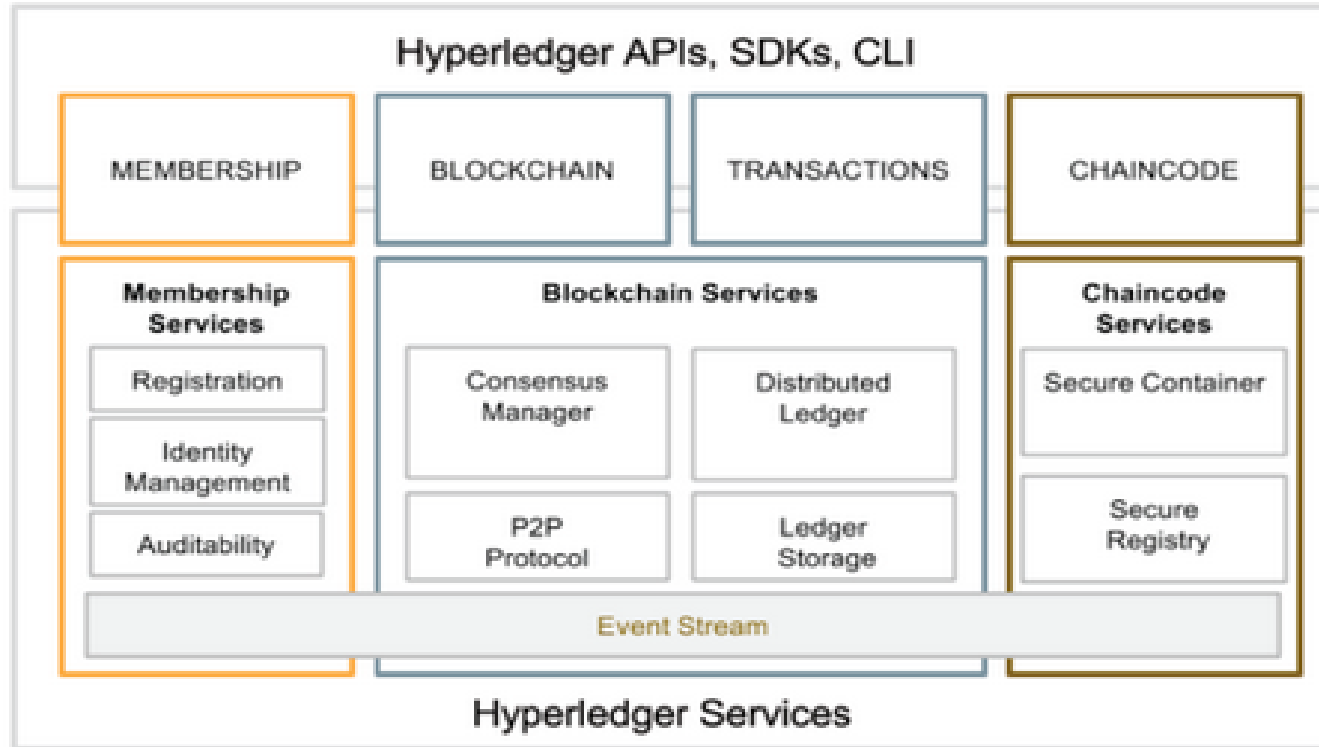
- Support a wide variety of industrial use cases with different requirements
- Comply with regulatory regimes that exist today
- Support
 - verified identities, private and confidential transactions
 - permissioned, shared ledger
 - performance, scaling, auditability, identity, security, and privacy
 - reduction of costly computations involved in proof of work

Hyperledger's Blockchain fabric use the following concepts as its underpinnings:

- Smart contracts
- Digital assets
- System of record repositories/stores
- Decentralized consensus-based network
- Pluggable consensus algorithms/models
- Cryptographic security

<https://github.com/OpenAssets/open-assets-protocol>

The Hyperledger project



The Hyperledger project

FINANCE

Hyperledger and its members are active defining financial services use cases and pursuing various finance PoCs.

LEARN MORE



HEALTHCARE

Hyperledger is committed to helping the healthcare industry realize the full potential of open source blockchain technologies.

LEARN MORE



SUPPLY CHAIN

Coming Soon

LEARN MORE



<https://www.hyperledger.org/industries>

Concepts and technologies

https://www.ibm.com/blockchain/what_is_blockchain.html

Before

Asset ownership and transfer between businesses is currently inefficient, slow, costly and vulnerable to manipulation. Everyone has their own ledger where discrepancies between business parties can increase settlement times.

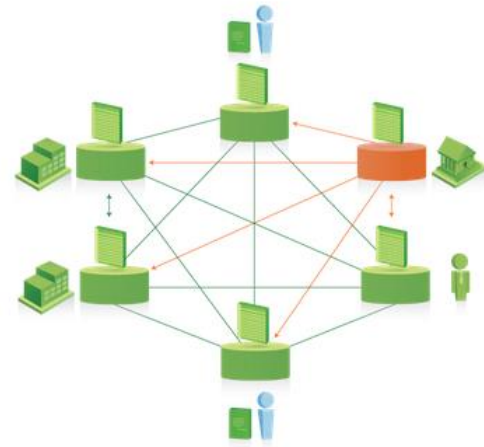
A new way is needed for Internet-age market enablement.

Concepts and technologies

https://www.ibm.com/blockchain/what_is_blockchain.html

Before

Asset ownership and transfer between businesses is currently inefficient, slow, costly and vulnerable to manipulation. Everyone has their own ledger where discrepancies between business parties can increase settlement times. A new way is needed for Internet-age market enablement.

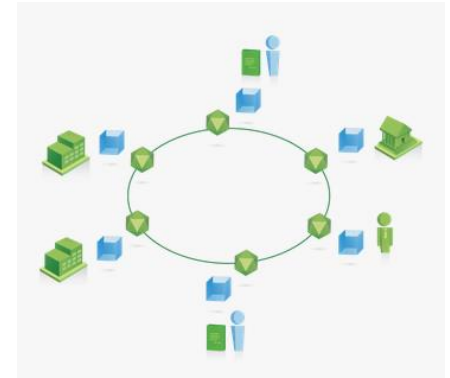


Concepts and technologies

https://www.ibm.com/blockchain/what_is_blockchain.html

After

- Blockchain technologies can be used to share a ledger across
- the business network.
- The network will be private to the parties concerned, permissioned so only authorized parties are allowed to join
- Secure using cryptographic technology to ensure that participants only see what they are allowed to see.
- The shared ledger will be more robust, since it is replicated and distributed.
- All transactions against the ledger will require consensus across the network, where provenance of information is clear and transparent.
- Transactions will be immutable (unchangeable) and final.



Concepts and technologies: Shared ledger

https://www.ibm.com/blockchain/what_is_blockchain.html

Shared ledger

It can act as a source of truth for businesses doing transactions on a blockchain:

- Records all transactions across the business network
- Is shared among participants
- Is replicated so each participant has their own copy
- Is permissioned, so participants see only appropriate transactions

Often, companies have multiple ledgers for multiple business networks in which they participate. It can be used for recording and totaling financial transactions.

Concepts and technologies: Smart contracts

https://www.ibm.com/blockchain/what_is_blockchain.html

Smart contracts

A smart contract can include a digital asset which is anything that has an owner and can be converted into value. Digital assets can be tangible or intangible. A smart contract can also include a digital representation of a set of business rules:

- Is embedded in the blockchain
- Is executed in a transaction
- Is verifiable, signed, and encoded in a programming language
- For example, it defines conditions under which corporate bond transfer occurs

Concepts and technologies: Consensus

https://www.ibm.com/blockchain/what_is_blockchain.html

Consensus

Entries in the ledger are synchronized to all ledgers in the network. Consensus ensures that these shared ledgers are exact copies, and lowers the risk of fraudulent transactions since tampering would have to occur across many places at the exact same time.

- All parties agree to the transaction and validate it via the peer network.
- Rules can also be established to validate transactions.
- This trusted and trustless participation makes commitment possible at a low cost.

IBM Blockchain uses a “pluggable” consensus system to meet the needs of different industry segments.

Concepts and technologies: Privacy and Confidentiality

https://www.ibm.com/blockchain/what_is_blockchain.html

Privacy and confidentiality

- Ability to protect records with a personal digital signature —
 - the blockchain generates a private and public key to seal that record.
- It is encrypted, hashed, and sent to the network of validating nodes
- Unique IDs for customer, invoice and reference numbers
- Although the ledger is shared, sometimes participants require:
 - Private transactions
- Identities that cannot be linked to specific transactions
- Transactions need to be authenticated, and cryptography is central to these processes.

Concepts and technologies: Privacy and Confidentiality

https://www.ibm.com/blockchain/what_is_blockchain.html

Privacy and confidentiality

- Ability to protect records with a personal digital signature —
 - the blockchain generates a private and public key to seal that record.
- It is encrypted, hashed, and sent to the network of validating nodes
- Unique IDs for customer, invoice and reference numbers
- Although the ledger is shared, sometimes participants require:
 - Private transactions
- Identities that cannot be linked to specific transactions
- Transactions need to be authenticated, and cryptography is central to these processes.

Concepts and technologies: Chaincode

<https://github.com/IBM-Blockchain/learn-chaincode>

Chaincode is Go (GoLang) or Java code that enables users to interact with a blockchain network. Whenever you 'invoke' a transaction on the network, you are calling a function in chaincode that reads and writes values to the ledger.

Chaincode Hello World

<https://github.com/ALT-F1/chaincode-helloworld/blob/master/helloworld.go>

```
/*
The MIT License (MIT)
*/

....
// HelloWorld example simple Chaincode implementation
type HelloWorld struct {
}

// Init is called during Deploy transaction after the container has been
func (t *HelloWorld) Init(stub *shim.ChaincodeStub, function string, args []string) ([]byte, error) {
    ....
}
// Invoke is called for every Invoke transactions. The chaincode may change its state variables
func (t *HelloWorld) Invoke(stub *shim.ChaincodeStub, function string, args []string) ([]byte, error) {
    fmt.Printf("HelloWorld in Invoke with function %s!\n", function)
    if function != "invoke" {
        return nil, errors.New("Invalid invoke function name. Expecting \"invoke\"")
    }
    .....}
}
```

Chaincode Hello World

<https://github.com/ALT-F1/chaincode-helloworld/blob/master/helloworld.go>


```
// Query is called for Query transactions. The chaincode may only read
// (but not modify) its state variables and return the result
func (t *HelloWorld) Query(stub *shim.ChaincodeStub, function string, args []string) ([]byte, error) {
    myLogger.Debug("HelloWorld Query is called!\n")
    if function != "query" {
        return nil, errors.New("Invalid query function name. Expecting \"query\"")
    }
    .....
}

func main() {
    myLogger.Debug("HelloWorld Main\n")
    err := shim.Start(new(HelloWorld))
    if err != nil {
        fmt.Printf("Error starting HelloWorld chaincode: %s\n", err)
    }
}
```

Section 3:

Blockchain on Bluemix

Blockchain in Bluemix Catalog

 IBM Bluemix Catalog

259

CatalogSupportAccount

All Categories (1) >

Infrastructure

Compute

Storage

Network

Security

Apps


Boilerplates


Cloud Foundry Runtimes

Containers

OpenWhisk

Mobile

 Blockchain


Filter 

Showing 1 result for "Blockchain" in 1 category

Services


Application Services

Deliver new web and mobile apps.

 **Blockchain**

Utilize IBM's Blockchain Technology within

IBM

 IBM Bluemix Catalog

259

CatalogSupportAccount

[View All](#)

Blockchain

Blockchain is a peer-to-peer distributed ledger technology for a new generation of transactional applications that establishes trust, accountability and transparency while streamlining business processes. Think of it as an operating system for interactions, with the potential to vastly reduce the cost and complexity of getting things done. The distributed ledger makes it easier to create cost-efficient business networks where virtually anything of value can be tracked and traded, without requiring a central point of control. This service is built on top of the Linux Foundation's

Service name:

Blockchain-p7

Credential name:

Credentials-1

Features

- **Spin up a test Blockchain Network in one click**
Spend less time creating and managing a blockchain network and more time
- **Membership services**
Take advantage of our first implementation of the membership services module, which encompasses many of the latest advances in

Blockchain in Bluemix Catalog

Blockchain

Blockchain is a peer-to-peer distributed ledger technology for a new generation of transactional applications that establishes trust, accountability and transparency while streamlining business processes. Think of it as an operating system for interactions, with the potential to vastly reduce the cost and complexity of getting things done. The distributed ledger makes it easier to create cost-efficient business networks where virtually anything of value can be tracked and traded, without requiring a central point of control. This service is built on top of the Linux Foundation's Hyperledger Project open source code. IBM has recently launched a new plan for High Security business networks which features a 4 node network running on dedicated infrastructure.

IBM

Connect to:

Leave unbound

[View Docs](#)

AUTHOR IBM
PUBLISHED 10/20/2016
TYPE Service
LOCATION US South

Service name:

Blockchain-w9

Credential name:

Credentials 1

Features

- **Spin up a test Blockchain Network in one click**

Spend less time creating and managing a blockchain network and more time focusing on writing your applications.

- **Create confidential digital assets**

Create digital transactions in your test applications that are processed quickly and securely over your permissioned network.

- **Membership services**

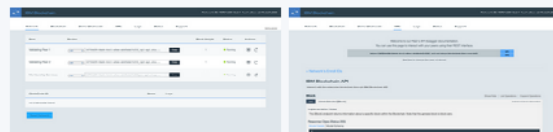
Take advantage of our first implementation of the membership services module, which encompasses many of the latest advances in cryptography.

- **Work with chaincode**

Smart contracts, written in chaincode, contain embedded business logic that allows you to define assets and write transaction instructions.

Images

Click an image to enlarge and view screen captures, slides, or videos. Screen caps show the user interface for the service after it has been provisioned.



Blockchain in Bluemix Catalog

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_overview.html

IBM Blockchain

Documentation

About blockchain

Last updated: 23 September 2016 | [Edit In GitHub](#)

What is blockchain?

Blockchain is a technology for a new generation of transactional applications that establishes trust, accountability and transparency, while streamlining business processes. The blockchain network was first introduced by bitcoin, but its practical uses extend far beyond cryptocurrency exchanges. With blockchain, IBM is reimagining the most fundamental business exchanges, and opening the door to a new world of digital interactions.

Blockchain is projected to vastly reduce the cost and complexity of cross-enterprise business processes. Its distributed ledger makes it easier to create cost-efficient business networks, where virtually anything of value can be tracked and traded, without a centralized point of control.

Blockchain is already showing great promise across a broad range of business applications. As just one example, blockchain networks allow securities trades to be settled in minutes, rather than days. Blockchain is also helping companies streamline the flow of goods and payments, and enabling manufacturers to reduce product recalls by openly sharing production logs with OEMs and regulators.

Getting started

What you need to know

› [About blockchain](#)

› Network plans

› HFC SDK for Node.js

› Testing blockchain networks

› Dashboard monitor

› **Samples and tutorials**

Getting support

Blockchain in Bluemix Catalog – sample apps the tutorials

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

Sample apps and tutorials

Last updated: 5 October 2016 | [Edit In GitHub](#)

The following samples demonstrate how applications and chaincode function in an IBM Blockchain network. To learn more about the Hyperledger Fabric v0.5 code, which underpins your blockchain network, visit the [Fabric Docs](#) section of the Linux Foundation's Hyperledger Project.

To experience chaincode applications in action, you can immediately deploy the Marbles, Commercial Paper or Car Lease demo below (click a Deploy to Bluemix button). Or continue reading to explore the Hello Chaincode tutorial.

-  **Marbles**
-  **Commercial Paper**
-  **Car Lease**

Using the Hello Chaincode tutorial

This tutorial guides you through using basic building blocks to code an elementary chaincode application. You will incrementally build a working chaincode that creates generic assets for exchanging on a network. Then you will interact with your chaincode through the network API. After completing this tutorial, you will be able to answer the following questions:

Getting started

What you need to know

› About blockchain

› Network plans

› HFC SDK for Node.js

› Testing blockchain network

› Dashboard monitor

▼ [Samples and tutorials](#)

Using the chaincode tutorial

Requirements for demos

Using Marbles demo

Using Commercial Paper demo

Using Car Lease demo

Non-deterministic chaincode

Getting support

Lab 1 Marbles

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

View your app

Sample apps and tutorials

Last updated: 5 October 2016 | [Edit In GitHub](#)


The following samples demonstrate how applications and chaincode function in an IBM Blockchain network. To learn more about the Hyperledger Fabric v0.5 code, which underpins your blockchain network, visit the [Fabric Docs](#) section of the Linux Foundation's Hyperledger Project.

To experience chaincode applications in action, you can immediately deploy the Marbles, Commercial Paper or Car Lease demo below (click a Deploy to Bluemix button). Or continue reading to explore the Hello Chaincode tutorial.

-  **Marbles**
-  **Commercial Paper**
-  **Car Lease**

Deploy this application to Bluemix

Deploying this app will create a private DevOps Services project for you. [Learn more.](#)



MARBLES
GIT URL: <https://github.com/ibm-blockchain/marbles.git>
GIT BRANCH: master

APP NAME
marbles-lennartf-1321

REGION	ORGANIZATION	SPACE
IBM Bluemix US S...	all@us.ibm.com	dev

DEPLOY

📢 New! The toolchains beta feature is now available. [Deploy this example using toolchains!](#)


Lab 1 Marbles – View Your App

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

View your app

Deploy this application to Bluemix

Deploying this app will create a private DevOps Services project for you. [Learn more](#)



MARBLES
GIT URL: <https://github.com/ibm-consulting-blockchain/marbles.git>
GIT BRANCH: master

- ✓ Created project successfully
- ✓ Cloned repository successfully
- ✓ Configured pipeline successfully
- ✓ Deployed to Bluemix successfully

Success!

You've added an instance of this app to your organization in Bluemix.

[VIEW YOUR APP](#)[EDIT CODE](#)


MARBLES P1HOMECREATE

Marbles

TIME 10/27/2016 08:20PM UTC

Bob'sLeroy's

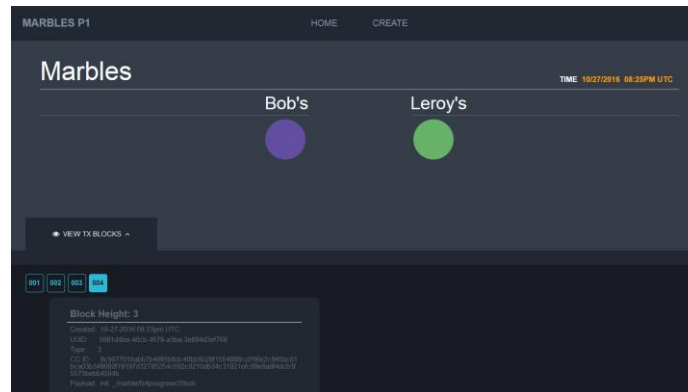
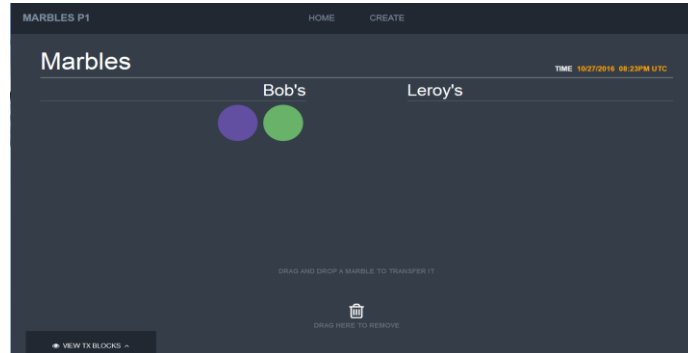
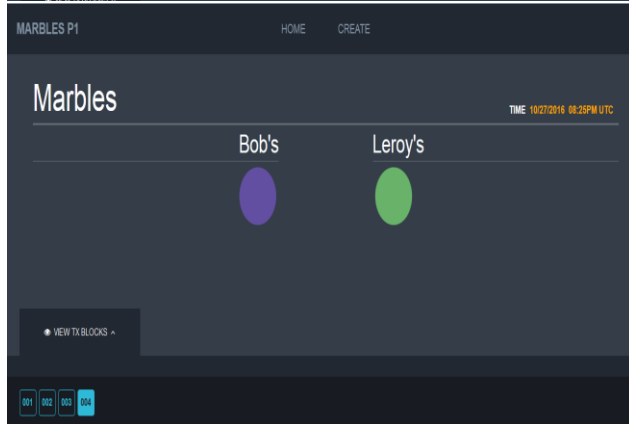
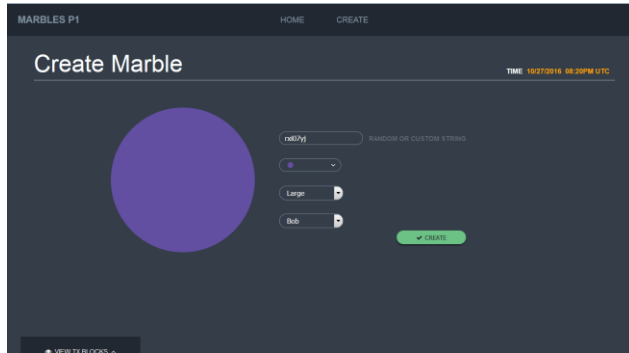
DRAG AND DROP A MARBLE TO TRANSFER IT


DRAG HERE TO REMOVE

Lab 1 Marbles – View Your App

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

View your app



Lab 1 Marbles – Edit Code

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

The screenshot shows the IBM Bluemix DevOps Services interface. The top navigation bar includes 'DASHBOARD', 'MY PROJECTS', 'EXPLORE', 'HELP', 'BLOG', and 'COMMUNITY'. Below this is a toolbar with 'File', 'Edit', 'View', 'Tools', and a dropdown menu showing 'marbles-lennartf-1353 (running: normal)'. There are also buttons for 'Live Edit', 'EDIT CODE', 'TRACK & PLAN', and 'BUILD & DEPLOY'. The left sidebar shows a file explorer for the project 'lennartf | marbles-lennartf-1353', listing files like '.bluemix', '.git', 'doc_images', 'launchConfigurations', 'public', 'routes', 'scss', 'utils', 'views', '.cfignore', '.gitignore', '.jshintignore', '.jshintrc', 'app.js', 'busters_css.json', 'busters_js.json', 'gulpfile.js', 'i_lost_my_marbles.md', 'LICENSE', 'manifest.yml', 'marbles_demo_notes.md', 'mycreds.json', 'package.json', 'README.md', 'setup.js', 'tutorial_part1.md', and 'tutorial_part2.md'. The main content area displays the 'README.md' file for the 'Marbles Demo'. It features a 'Deploy to Bluemix' button and an 'Application Background' section. The background text explains that the application demonstrates transferring marbles between two users using IBM Blockchain, Node.js, and GoLang. It describes the chaincode's role in creating and storing marbles. A code block lists the attributes of a marble: name, color, size, and user. The text continues to describe the Web UI and the use of the IBM Blockchain SDK. A 'Tutorial / Documentation' section provides links to chaincode documentation and Marbles tutorials. A 'Projects Contents' section lists the URLs for the Marbles Part 1 and Part 2 tutorials. Finally, a 'Privacy Notice' section is mentioned.

IBM Bluemix DevOps Services

DASHBOARD MY PROJECTS EXPLORE HELP BLOG COMMUNITY

File Edit View Tools marbles-lennartf-1353 (running: normal) Live Edit EDIT CODE TRACK & PLAN BUILD & DEPLOY

lennartf | marbles-lennartf-1353

▼ README.md

Marbles Demo

[Deploy to Bluemix](#)

Application Background

Hold on to your hats everyone, this application is going to demonstrate transferring marbles between two users leveraging IBM Blockchain. We are going to do this in Node.js and a bit of GoLang. The backend of this application will be the GoLang code running in our blockchain network. The chaincode itself will create a marble by storing it to the chaincode state. The chaincode itself is able to store data as a string in a key/value pair setup. Thus we will stringify JSON objects to store more complex structures.

Attributes of a marble:

```
1. name (unique string, will be used as key)
1. color (string, css color names)
1. size (int, size in mm)
1. user (string)
```

We are going to create a Web UI that can set these values and pass them to the chaincode. Interacting with the chaincode is done with a HTTP REST call to a peer on the network. The ibc-js SDK will abstract the details of the REST calls away. This allow us to use dot notation to call our GoLang functions (such as `chaincode.init_marble(args)`).

Start the tutorials below to have your own marbles blockchain demo!

Tutorial / Documentation

- Looking for chaincode documentation? Check out the [learn chaincode](#) repo - start here!
- Tutorial for Marbles [Part 1](#)
- Tutorial for Marbles [Part 2](#)
- Documentation for IBM Blockchain [IBC-JS SDK](#) (our REST based SDK)

Projects Contents

If you run marbles on local host you will have these two urls:

- Marbles Part 1 - <http://localhost:3000/p1>
- Marbles Part 2 - <http://localhost:3000/p2>

Privacy Notice

Lab 1 Marbles – Bluemix console

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

manifesttest.yml		128	1	1	 Running		
marbles-lennartf-1317	marbles-lennartf-1317.mybluemix.net	512	1	1	 Running	 	
MobileFoundation-r...	mobilefoundation-rk-fe-server.myblu...	1024	1	1	 Running	 	
NodeRedNI	NodeRedNI.mybluemix.net	512	1	1	 Running	 	
outthink	outthink.mybluemix.net	128	1	1	 Running	 	

Lab 1 Marbles – Bluemix console

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

The screenshot shows the Bluemix console interface for an application named 'marbles-lennartf-1317'. The status is 'your app is running'. The main dashboard includes four metrics: Buildpack (SDK for Node.js™), Instances (1, All instances are running, Health is 100%), MBs per Instance (512), and Total MB Allocation (512, 640 MBs still available). Below these are sections for Connections, Runtime Cost (\$0.49), Activity Log, and Continuous Delivery. The Activity Log shows a crash and a start event. The Continuous Delivery section has a 'GIT URL' and two buttons, 'Configure' and 'Edit Code', which are circled in red.

marbles-lennartf-1317 Status: ● your app is running View App

Runtime

- BUILDPACK**
SDK for Node.js™
- INSTANCES**
1
All instances are running
Health is 100%
- MBs PER INSTANCE**
512
- TOTAL MB ALLOCATION**
512
640 MBs still available

Connections

No services are connected to this app
You can create or bind a service:

[Connect New](#) [Connect Existing](#)

Runtime Cost

\$0.49
Current Charges for Billing Period

\$0.49
Estimated Total for Billing Period
(10/01-10/31)

Current and estimated cost excludes [connected services](#).

[View Full Usage Details](#)

Activity Log

- an instance of the app crashed: out of memory
exit status: 255, CRASHED
10/28/2016 2:24 AM | marbles-lennartf-1317
- started marbles-lennartf-1317 app
10/27/2016 1:19 PM | al@us.ibm.com
- updated module: lennartf-1317

Continuous Delivery

GIT URL
<https://hub.jazz.net/git/lennartf/marbles-lennartf-1317>

[Configure](#) [Edit Code](#)

Lab 1 Marbles – DevOps Services Build and Deploy

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

The screenshot displays the IBM Bluemix DevOps Services console interface. At the top, the navigation bar includes links for DASHBOARD, MY PROJECTS, EXPLORE, HELP, BLOG, and COMMUNITY. The main header shows the project name 'lennartf | marbles-lennartf-1317' and action buttons for EDIT CODE, TRACK & PLAN, and BUILD & DEPLOY.

The central section features a video player titled 'Build & Deploy Pipeline Core Concepts' with a duration of 2:58. To the right of the video, a welcome message states: 'Welcome to the Build & Deploy Pipeline. If you want to automate your continuous deployments, set up stages to retrieve input and run jobs, such as builds, tests, and deployments. We've configured the first two stages and have deployed your sample app for you. Pushing changes to the repository automatically trigger a new build and then a deployment. To learn more about using IBM Bluemix with IBM Bluemix DevOps Services together for development and continuous integration, check out the [Build & Deploy Docs](#)'.

Below the video, the 'Pipeline: All Stages' section shows two stages:

- Empty Build Stage**: Indicated as 'STAGE PASSED'. The last input is a Git URL commit by Lennart Fran... 14 hr ago, imported from <https://github.com/ibm-b...>. The job 'Empty Build...' is shown as 'Succeeded 14 hr ago'. The last execution result is 'Empty Build Job 1'.
- Deploy to Bluemix**: Indicated as 'STAGE PASSED'. The last input is 'Stage: Empty Build Stage /...'. The job 'Push to Clo...' is shown as 'Succeeded 14 hr ago'. The last execution result shows the application 'marbles-lennartf-1317' deployed to marbles-lennartf-1317.mybluemix.net, with a link to [View runtime log](#).

A '+ ADD STAGE' button is available to the right of the stage cards.

Lab 1 Marbles – DevOps Services Edit Code – Build and Deploy

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

IBM Bluemix DevOps Services

DASHBOARD MY PROJECTS EXPLORE HELP ▾ BLOG COMMUNITY

File Edit View Tools ● marbles-lennartf-1317 (running: normal) Live Edit. EDIT CODE TRACK & PLAN BUILD & DEPLOY

lennartf | marbles-lennartf-1317

▼ README.md

Marbles Demo

Deploy to Bluemix

Application Background

Hold on to your hats everyone, this application is going to demonstrate transferring marbles between two users leveraging IBM Blockchain. We are going to do this in Node.js and a bit of GoLang. The backend of this application will be the GoLang code running in our blockchain network. The chaincode itself will create a marble by storing it to the chaincode state. The chaincode itself is able to store data as a string in a key/value pair setup. Thus we will stringify JSON objects to store more complex structures.

Attributes of a marble:

```
1. name (unique string, will be used as key)
1. color (string, css color names)
1. size (int, size in mm)
1. user (string)
```

We are going to create a Web UI that can set these values and pass them to the chaincode. Interacting with the chaincode is done with a HTTP REST call to a peer on the network. The ibc-js SDK will abstract the details of the REST calls away. This allows us to use dot notation to call our GoLang functions (such as `chaincode.init_marble(args)`).

Start the tutorials below to have your own marbles blockchain demo!

Tutorial / Documentation

- Looking for chaincode documentation? Check out the [learn chaincode](#) repo - start here!
- Tutorial for Marbles [Part 1](#)
- Tutorial for Marbles [Part 2](#)
- Documentation for IBM Blockchain [IBC-JS SDK](#) (our REST based SDK)

Projects Contents




If you run marbles on local host you will have these two urls:

1. Marbles Part 1 - <http://localhost:3000/p1>
2. Marbles Part 2 - <http://localhost:3000/p2>

Privacy Notice

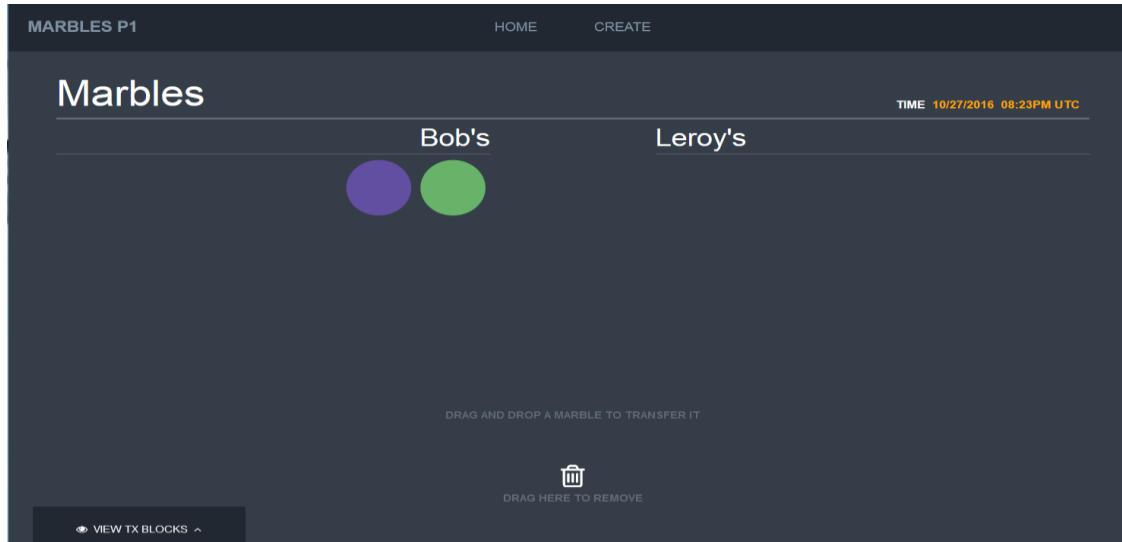
Lab 1 Marbles – Console

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

manifesttest.yml		128	1	1	 Running		
marbles-lennartf...	marbles-lennartf-1317.mybluemix..	512	1	1	 Running	 	
MobileFoundatio...	mobilefoundation-rk-fe-server.m...	1024	1	1	 Running	 	
NodeRedNI	NodeRedNI.mybluemix.net	512	1	1	 Running	 	

Lab 1 Marbles – View Your App

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html




Lab 2 Commercial Paper

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

View your app

Deploy this application to Bluemix

Deploying this app will create a private DevOps Services project for you. [Learn more.](#)



CP-WEB
GIT URL: <https://github.com/IBM-Blockchain/cp-web.git>
GIT BRANCH: master

APP NAME

REGION

ORGANIZATION

SPACE

IBM Bluemix US S...

alf@us.ibm.com

dev


DEPLOY

i Now! The **toolchains** beta feature is now available. [Deploy this example using toolchains!](#)

You are logged in as [alf@us.ibm.com](#). [Log out.](#)
[Terms of Use](#)

Deploy this application to Bluemix

Deploying this app will create a private DevOps Services project for you. [Learn more.](#)



CP-WEB
GIT URL: <https://github.com/IBM-Blockchain/cp-web.git>
GIT BRANCH: master

✓ Created project successfully

✓ Cloned repository successfully

✓ Configured pipeline successfully

✓ Deployed to Bluemix successfully

i Now! The **toolchains** beta feature is now available. [Deploy this example using toolchains!](#)

Success!

You've added an instance of this app to your organization in Bluemix.


VIEW YOUR APP

EDIT CODE

You are logged in as [alf@us.ibm.com](#). [Log out.](#)
[Terms of Use](#)

Lab 2 Commercial Paper – View Your App

COMMERCIAL PAPER

 LOGIN ▾

Login

Username

username

Password

Login

Register

Username

username


Register

User roles are assigned through tags in the user name. Include the string 'auditor' in your user name to register as that user type. Everyone else is will be considered a regular user. Updates to the chaincode will remove this limitation in the future.

Lab 2 Commercial Paper – View Your App – Trade Center

COMMERCIAL PAPER

AUDIT

 AUDITOR ▾


Trade Center

TIME 10/28/2016 09:02AM UTC

Account Balance:Loading from blockchain...

Filter Trades

DATE↑	CUSIP	TICKER	PAR	QTY	DISCOUNT	MATURITY	ISSUER	OWNER	ACTION
-------	-------	--------	-----	-----	----------	----------	--------	-------	--------

 VIEW TX BLOCKS ^

Lab 2 Commercial Paper in Bluemix Dashboard

bluemixintro1	bluemixintro1.mybluemix.net	128	1	1	 Running	  
bluemixintroiot	bluemixintroiot.mybluemix.net	512	1	1	 Running	  
cp-web-lennartf...	cp-web-lennartf-1415.mybluemix...	512	1	1	 Running	  
falkenberg	falkenberg.mybluemix.net	128	1	0	 Stopped	 
IndiaHackathon...	IndiaHackathonMFSS.mybluemix...	512	1	1	 Running	  
manifesttest.yml		128	1	1	 Running	 

Lab 2 Commercial Paper in Bluemix Dashboard

The screenshot shows the IBM Bluemix Cloud Foundry Applications dashboard for the application **cp-web-lennartf-1415**. The status is "Your app is running".

Runtime Metrics:

- BUILDPACK:** SDK for Node.js™
- INSTANCES:** 1 (All instances are running, Health is 100%)
- MBs PER INSTANCE:** 512
- TOTAL MB ALLOCATION:** 512 (640 MBs still available)

Connections (1): cpblockchain

Runtime Cost:

- Current Charges for Billing Period:** \$0.41
- Estimated Total for Billing Period (10/01-10/31):** \$0.41
- Current and estimated cost excludes connected services.
- [View Full Usage Details](#)

Activity Log:

- started cp-web-lennartf-1415 app
10/27/2016 2:25 PM | all@us.ibm.com
- updated cp-web-lennartf-1415 app
changed routes
10/27/2016 2:25 PM | all@us.ibm.com

Continuous Delivery:

- GIT URL:** <https://github.com/jazz.net/gf/lennartf/cp-web-lennartf-1415>
- [Configure](#)
- [Edit Code](#)

Lab 2 Commercial Paper – DevOps Services – Configure - Build and Deploy

IBM Bluemix DevOps Services

DASHBOARD MY PROJECTS EXPLORE HELP ▾ BLOG COMMUNITY

lennartf | cp-web-lennartf-1415

EDIT CODE TRACK & PLAN BUILD & DEPLOY

Pipeline: All Stages

Empty Build Stage

STAGE PASSED

LAST INPUT [Git URL](#)

Last commit by Lennart Fran... 12 hr ago
[Imported from https://github.com/IBM-B...](#)

JOBS [View logs and history](#)

✓ Empty Build... Succeeded 12 hr ago

LAST EXECUTION RESULT

Empty Build Job 1

Deploy to Bluemix

STAGE PASSED

LAST INPUT Stage: Empty Build Stage /...

Empty Build Job 1

JOBS [View logs and history](#)

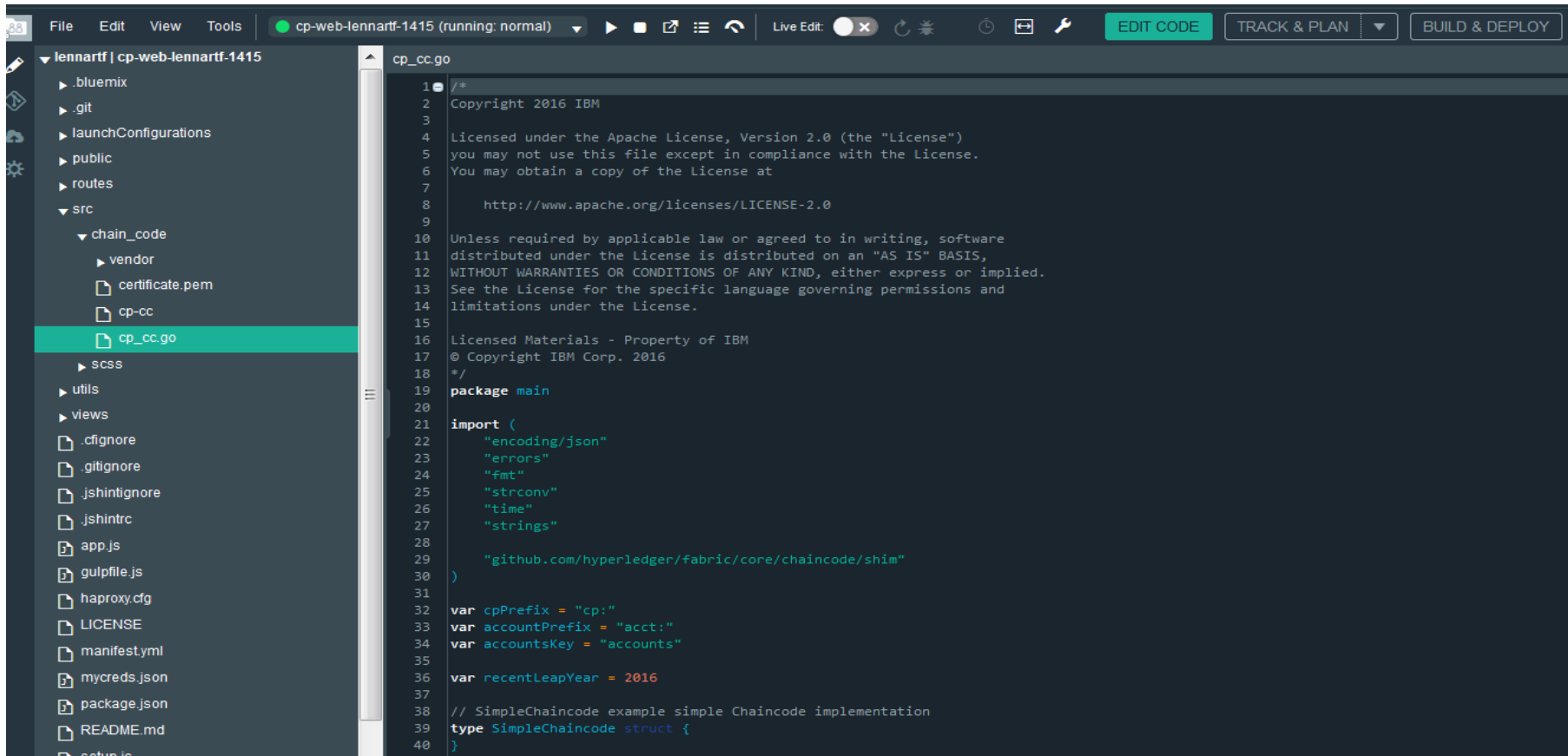
✓ Push to Clo... Succeeded 12 hr ago

LAST EXECUTION RESULT

Loading...

+ ADD STAGE





















Lab 2 Commercial Paper – Devops Services – Edit Code – Build and Deploy



The screenshot displays a web-based IDE interface. The top bar shows the project name 'cp-web-lennartf-1415 (running: normal)' and buttons for 'EDIT CODE', 'TRACK & PLAN', and 'BUILD & DEPLOY'. The left sidebar shows a file explorer for the project 'lennartf | cp-web-lennartf-1415'. The file 'cp_cc.go' is selected and highlighted in green. The main editor area shows the code for 'cp_cc.go'.

```
1 /*
2 Copyright 2016 IBM
3
4 Licensed under the Apache License, Version 2.0 (the "License")
5 you may not use this file except in compliance with the License.
6 You may obtain a copy of the License at
7
8     http://www.apache.org/licenses/LICENSE-2.0
9
10 Unless required by applicable law or agreed to in writing, software
11 distributed under the license is distributed on an "AS IS" BASIS,
12 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 See the License for the specific language governing permissions and
14 limitations under the License.
15
16 Licensed Materials - Property of IBM
17 © Copyright IBM Corp. 2016
18 */
19 package main
20
21 import (
22     "encoding/json"
23     "errors"
24     "fmt"
25     "strconv"
26     "time"
27     "strings"
28
29     "github.com/hyperledger/fabric/core/chaincode/shim"
30 )
31
32 var cpPrefix = "cp:"
33 var accountPrefix = "acct:"
34 var accountsKey = "accounts"
35
36 var recentLeapYear = 2016
37
38 // SimpleChaincode example simple Chaincode implementation
39 type SimpleChaincode struct {
40     }
```



Lab 2 Commercial Paper in Bluemix Dashboard – View App

bluemixintro1	bluemixintro1.mybluemix.net	128	1	1	 Running	  
bluemixintroiot	bluemixintroiot.mybluemix.net	512	1	1	 Running	  
cp-web-lennartf...	cp-web-lennartf-1415.mybluemix...	512	1	1	 Running	  
falkenberg	falkenberg.mybluemix.net	128	1	0	 Stopped	 
IndiaHackathon...	IndiaHackathonMFSS.mybluemix...	512	1	1	 Running	  
manifesttest.yml		128	1	1	 Running	 

Lab 2 Commercial Paper – View Your App – Trade Center

COMMERCIAL PAPER

AUDIT

 AUDITOR ▾


Trade Center

TIME 10/28/2016 09:02AM UTC

Account Balance:Loading from blockchain...

Filter Trades

DATE↑	CUSIP	TICKER	PAR	QTY	DISCOUNT	MATURITY	ISSUER	OWNER	ACTION
-------	-------	--------	-----	-----	----------	----------	--------	-------	--------

 VIEW TX BLOCKS ^

Lab 3 Car Lease

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

Deploy this application to Bluemix

Deploying this app will create a private DevOps Services project for you. [Learn more.](#)



CAR-LEASE-DEMO

GIT URL: <https://github.com/IBM-Blockchain/car-lease-demo.git>
GIT BRANCH: master

APP NAME

car-lease-demo-lennartf-1339

REGION

IBM Bluemix US S...

ORGANIZATION

alf@us.ibm.com

SPACE

dev

DEPLOY

New! The **toolchains** beta feature is now available. [Deploy this example using toolchains!](#)

You are logged in as [alf@us.ibm.com](#). [Log out](#)
[Terms of Use](#)



CAR-LEASE-DEMO

GIT URL: <https://github.com/IBM-Blockchain/car-lease-demo.git>
GIT BRANCH: master

- Created project successfully
- Cloned repository successfully
- Configured pipeline successfully
- Deployed to Bluemix successfully

New! The **toolchains** beta feature is now available. [Deploy this example using toolchains!](#)

Success!

You've added an instance of this app to your organization in Bluemix.

[VIEW YOUR APP](#)

[EDIT CODE](#)

Lab 3 Car Lease - View App



BLOCKCHAIN CAR LEASING DEMO

Main Menu:

Welcome to the Car Leasing Demo.

To get a scenario set up click on the link to the admin console then use one of the Create Scenario buttons. This will create cars and move them to their locations.

Otherwise you can create your own cars by clicking on Create Asset.

Regulator

[Live Stats](#)

[Regulator View](#)

[Create Asset](#)

Transfer Asset

[Regulator → Manufacturer](#)

[Manufacturer → Dealership](#)

[Dealership → Lease Company](#)

[Lease Company → Lessee](#)

[Lessee → Scrap Merchant](#)

Update Asset

[Manufacturer Update](#)

Dispose Asset

[Scrap Merchant → Scrap](#)

Admin

[Admin Console](#)

Lab 3 Car Lease - View App

Admin Console

Demo setup:

Create Simple Scenario

Create Full Scenario

Creating Scenario

Creating vehicles ✓

Transferring vehicles to manufacturers ✓


Updating vehicles' details ✓

Transferring vehicles to private owners ✓

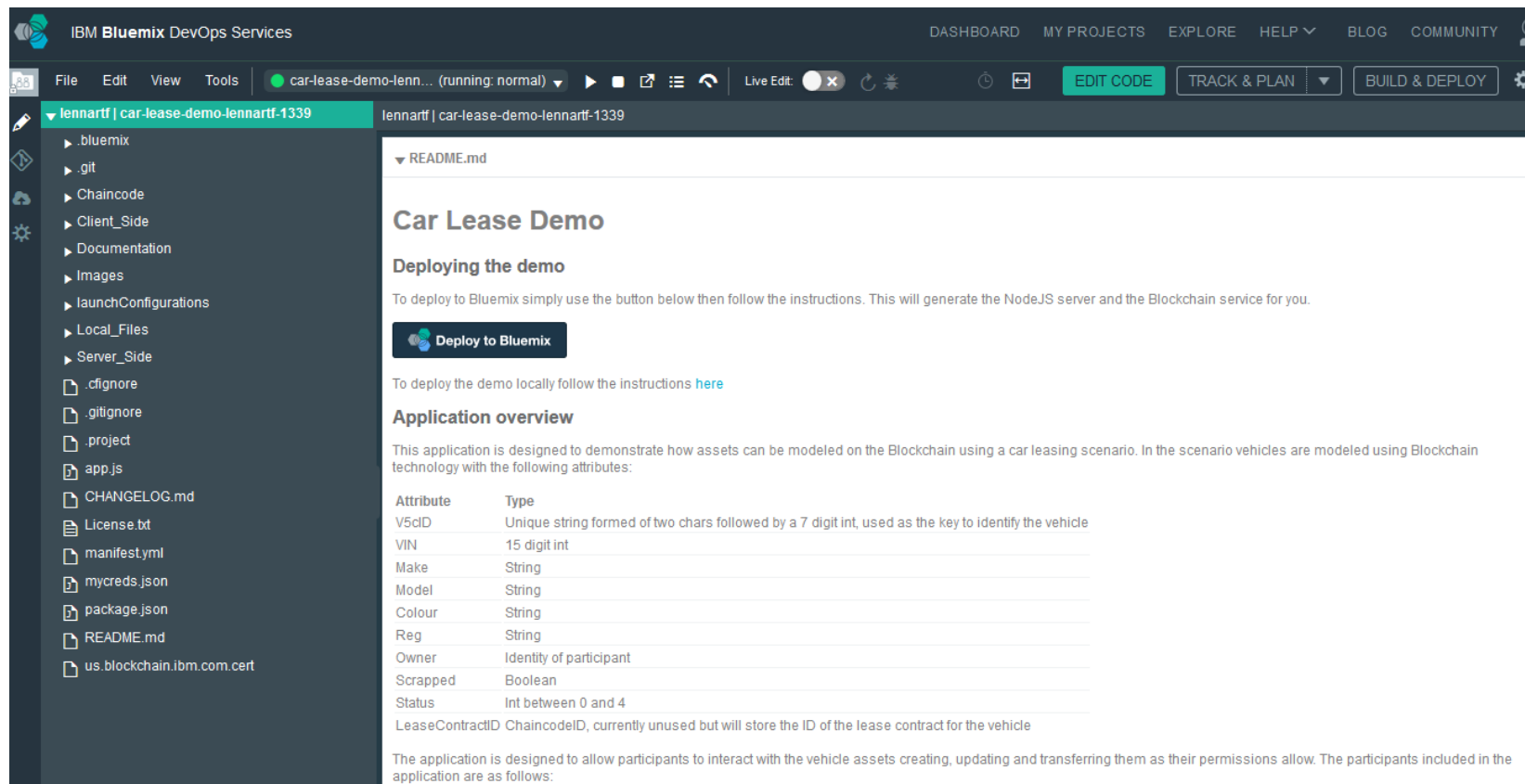
Demo setup ✓

OK

Lab 3 Car Lease - View App

	Scenario Creation Complete
Scenario	Scenario creation complete
	

Lab 3 Car Lease - Edit Code – Build and Deploy



IBM Bluemix DevOps Services

DASHBOARD MY PROJECTS EXPLORE HELP BLOG COMMUNITY

File Edit View Tools car-lease-demo-lenn... (running: normal) Live Edit: EDIT CODE TRACK & PLAN BUILD & DEPLOY

lennartf | car-lease-demo-lennartf-1339


lennartf | car-lease-demo-lennartf-1339

▼ README.md

Car Lease Demo

Deploying the demo

To deploy to Bluemix simply use the button below then follow the instructions. This will generate the NodeJS server and the Blockchain service for you.

 Deploy to Bluemix

To deploy the demo locally follow the instructions [here](#)

Application overview

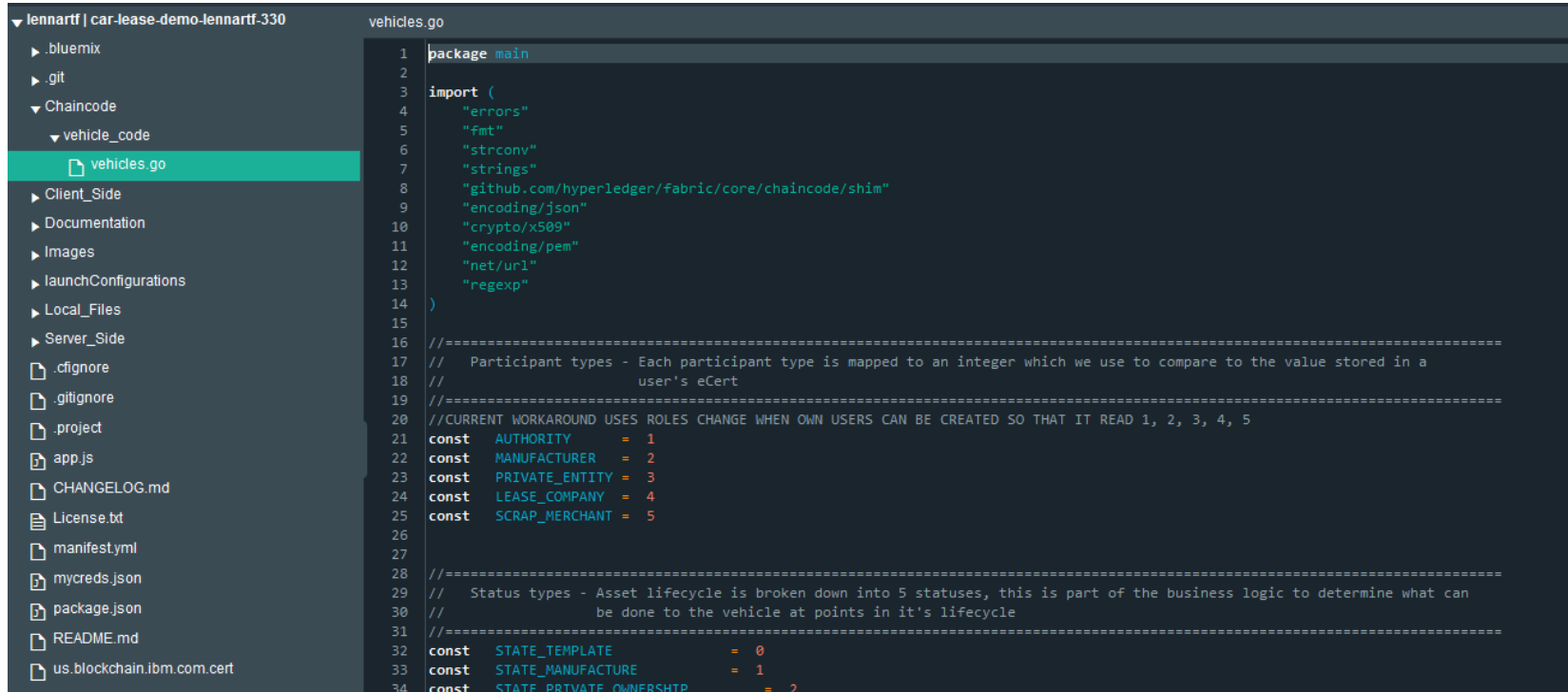
This application is designed to demonstrate how assets can be modeled on the Blockchain using a car leasing scenario. In the scenario vehicles are modeled using Blockchain technology with the following attributes:

Attribute	Type
V5clD	Unique string formed of two chars followed by a 7 digit int, used as the key to identify the vehicle
VIN	15 digit int
Make	String
Model	String
Colour	String
Reg	String
Owner	Identity of participant
Scrapped	Boolean
Status	Int between 0 and 4

LeaseContractID ChaincodeID, currently unused but will store the ID of the lease contract for the vehicle

The application is designed to allow participants to interact with the vehicle assets creating, updating and transferring them as their permissions allow. The participants included in the application are as follows:

Lab 3 Car Lease - Edit Code – Build and Deploy – Chain Code















The screenshot shows a code editor interface. On the left is a file explorer for a project named 'lennartf | car-lease-demo-lennartf-330'. The 'Chaincode' folder is expanded, showing 'vehicle_code' and 'vehicles.go' (which is selected). Other files in the explorer include 'Client_Side', 'Documentation', 'Images', 'launchConfigurations', 'Local_Files', 'Server_Side', '.cignore', '.gitignore', '.project', 'app.js', 'CHANGELOG.md', 'License.txt', 'manifest.yml', 'mycreds.json', 'package.json', 'README.md', and 'us.blockchain.ibm.com.cert'.





The main editor area displays the code for 'vehicles.go'. The code is as follows:

```
1 package main
2
3 import (
4     "errors"
5     "fmt"
6     "strconv"
7     "strings"
8     "github.com/hyperledger/fabric/core/chaincode/shim"
9     "encoding/json"
10    "crypto/x509"
11    "encoding/pem"
12    "net/url"
13    "regexp"
14 )
15
16 //=====
17 // Participant types - Each participant type is mapped to an integer which we use to compare to the value stored in a
18 //                      user's eCert
19 //=====
20 //CURRENT WORKAROUND USES ROLES CHANGE WHEN OWN USERS CAN BE CREATED SO THAT IT READ 1, 2, 3, 4, 5
21 const AUTHORITY = 1
22 const MANUFACTURER = 2
23 const PRIVATE_ENTITY = 3
24 const LEASE_COMPANY = 4
25 const SCRAP_MERCHANT = 5
26
27
28 //=====
29 // Status types - Asset lifecycle is broken down into 5 statuses, this is part of the business logic to determine what can
30 //                  be done to the vehicle at points in it's lifecycle
31 //=====
32 const STATE_TEMPLATE = 0
33 const STATE_MANUFACTURE = 1
34 const STATE_PRIVATE_OWNERSHIP = 2
```



Bluemix Console

bluemixintroiot	bluemixintroiot.mybluemix.net	512	1	1	 Running	  
car-lease-demo-	car-lease-demo-lennartf-330.mybluemix.net	512	1	1	 Running	  
cp-web-lennartf	cp-web-lennartf-1415.mybluemix.net	512	1	1	 Running	  


Bluemix Catalog

 car-lease-demo-lennartf-330 Status: ● Your app is running View App   


Runtime




BUILDPACK
SDK for Node.js™



INSTANCES
All instances are running
Health is 100%




MBS PER INSTANCE





TOTAL MB ALLOCATION
128 MBs still available ⓘ

Connections (1)

 car_lease_blockchain

Activity Log

-  started car-lease-demo-lennartf-330 app
10/28/2016 3:32 AM | alf@us.ibm.com
-  updated car-lease-demo-lennartf-330 app
changed routes
10/28/2016 3:31 AM | alf@us.ibm.com

Runtime Cost


\$0.00
Current Charges for Billing Period

\$0.00
Estimated Total for Billing Period
(10/01-10/31)

Current and estimated cost excludes connected services.

View Full Usage Details













Continuous Delivery

 **GIT URL**
<https://hub.jazz.net/git/lennartf/car-lease-demo-lennartf-330>

Configure

Edit Code

Bluemix Catalog

bluemixintroiot	bluemixintroiot.mybluemix.net	512	1	1	 Running			
car-lease-demo-...	car-lease-demo-lennartf-330.mybluemix.net	512	1	1	 Running			
cp-web-lennartf	cp-web-lennartf-1415.mybluemix.net	512	1	1	 Running			

Bluemix Catalog

IBM Bluemix Catalog

Categories (1) >


Structure

Showing 1 result for "Blockchain" in 1 category

Services

Application Services

Deliver new web and mobile apps.

 **Blockchain**

Utilize IBM's Blockchain Technology within Bluemix

IBM

Bluemix Catalog

Service name:

Blockchain-kl

Credential name:

Credentials-1

Features

- **Spin up a test Blockchain Network in one click**

Spend less time creating and managing a blockchain network and more time focusing on writing your applications.

- **Create confidential digital assets**

Create digital transactions in your test applications that are processed quickly and securely over your permissioned network.

- **Membership services**

Take advantage of our first implementation of the membership services module, which encompasses many of the latest advances in cryptography.

- **Work with chaincode**

Smart contracts, written in chaincode, contain embedded business logic that allows you to define assets and write transaction instructions.

ate Monthly Cost

[Calculator](#)

Create

← Application Services

Blockchain-kl

Manage

Service Credentials

Connections


Welcome to the Starter Developer Network on IBM Blockchain!

LAUNCH 

Welcome, alfi@us.ibm.com!

This service is intended for developers who consider themselves early adopters and want to get involved with IBM's approach to business networks that maintain, secure and share a replicated ledger using blockchain technology.

Bluemix Catalog

 IBM Blockchain

Network ID: 3c945fd9b0e645648ef133fd9f8f88d9

NetworkBlockchainDemo ChaincodeAPIsLogsStatusSupport

Peer

Members

Validation

Validation

Validation

Validation

Getting started

Getting Started

Welcome to your blockchain network's console page. From here you can do many things.
If this is your first time using IBM Blockchain you should deploy some chaincode. Our Demo Chaincode tab can help with that!

Network

Blockchain

Demo Chaincode

APIs

Logs

Deploy your first chaincode!

1) Go to "Demo Chaincode" and select a chaincode template. Then hit deploy.

2) Click the pre-built query/invoke buttons.

3) Take the sample to the next level by creating API calls in the "APIs" tab. You can get the chaincode ID from the networks tab.

Blockchain, Demo Chaincode

Network

Blockchain

Demo Chaincode

APIs

Logs

Status

Support

Not sure how to get started? Pick a demo and deploy its chaincode right from this page!

(Deploying will submit chaincode to your network. You will then be able to interact with the chaincode.)

Application

Description

Links

Interact

Example02

1/5 difficulty

Store two integers named A and B.
Subtract from one and add to the other.

[Chaincode](#)

[Show Actions](#)

Deploy

Marbles

2/5 difficulty

Create marble assets and trade them with
your friend Leroy.

[GitHub](#), [Chaincode](#), [Docs](#)

[Show Actions](#)

Deploy

**Commercial
Paper**

3/5 difficulty

Buy and sell business to business
monetary loans.

[GitHub](#), [Chaincode](#), [Docs](#)

[Show Actions](#)

Deploy

Bluemix chaincode samples

[Network](#)[Blockchain](#)[Demo Chaincode](#)[APIs](#)[Logs](#)[Status](#)[Support](#)

Not sure how to get started? Pick a demo and deploy its chaincode right from this page!

(Deploying will submit chaincode to your network. You will then be able to interact with the chaincode.)

Application	Description	Links	Interact
Example02 1/5 difficulty	Store two integers named A and B. Subtract from one and add to the other.	Chaincode	Show Actions
Marbles 2/5 difficulty	Create marble assets and trade them with your friend Leroy.	GitHub , Chaincode , Docs	Show Actions
Commercial Paper 3/5 difficulty	Buy and sell business to business monetary loans.	GitHub , Chaincode , Docs	Show Actions

Bluemix Example02 sample

Application	Description	Links	Interact
Example02 1/5 difficulty	Store two integers named A and B. Subtract from one and add to the other.	Chaincode	Show Actions <div>Deploy</div>
Select the correct chaincode		<div>example02: 72c3c395...</div>	<div>Transfer from A to B</div> <div>Transfer from B to A</div> <div>Query A</div> <div>Query B</div>

Chaincode: Marbles

Marbles

2/5 difficulty

Create marble assets and trade them with your friend Leroy.

[GitHub](#), [Chaincode](#), [Docs](#)

[Show Actions](#)

Deploy



Commercial

Paper

3/5 difficulty

Buy and sell business to business monetary loans.

[GitHub](#), [Chaincode](#), [Docs](#)

[Show Actions](#)


Deploy



Demo Chaincode successfully deployed:

Now you can hit an action button below "Actions" to invoke a chaincode function.



Show API details 

Blockchain Marbles

Marbles

2/5 difficulty

Create marble assets and trade them with your friend Leroy.

[GitHub](#), [Chaincode](#), [Docs](#)

[Show Actions](#)

Deploy

Select the correct chaincode

marbles: d827db00...



Create marble

Trade away
marble

Delete marble

Query marble

Bluemix Commercial Paper

Commercial
Paper

3/5 difficulty

Buy and sell business to business
monetary loans.

[GitHub](#), [Chaincode](#), [Docs](#)

[Show Actions](#)

Deploy



Commercial
Paper

3/5 difficulty

Buy and sell business to business
monetary loans.

[GitHub](#), [Chaincode](#), [Docs](#)

[Show Actions](#)

Deploy

Select the correct chaincode

cp: bc25e428... ▼

Register Accounts

Seed paper

Buy paper

Query account

Welcome to the Starter Developer Network on IBM Blockchain!

← Application Services

Blockchain-yk

Manage

Service Credentials

Connections

Welcome to the Starter Developer Network
on IBM Blockchain!

LAUNCH 

Welcome, alf@us.ibm.com!

This service is intended for developers who consider themselves early adopters and want to get involved with IBM's approach to business networks that maintain, secure and share a replicated ledger using blockchain technology.

Welcome to the Starter Developer Network on IBM Blockchain!

Getting Started

Welcome to your blockchain network's console page. From here you can do many things.
If this is your first time using IBM Blockchain you should [deploy some chaincode](#). Our Demo Chaincode tab can help with that!

<h3>Network</h3> <p>View the status of your network peers and deployed chain code</p>	<h3>Blockchain</h3> <p>Explore transactions written to the network</p>	<h3>Demo Chaincode</h3> <p>Deploy sample chain code templates in one-click</p>	<h3>APIs</h3> <p>Learn and interact with API docs</p>
---	--	--	---

Logs

View peer or membership service logs

[Deploy your first chaincode!](#)

- 1) Go to "Demo Chaincode" and select a chaincode template. Then hit deploy.
- 2) Click the pre-built query/invoke buttons.
- 3) Take the sample to the next level by creating API calls in the "APIs" tab. You can get the chaincode ID from the networks tab.

Welcome to the Starter Developer Network on IBM Blockchain!

Welcome to our Peer's API Swagger documentation.

You can use this page to interact with your peers using their REST interface.

`https://7aae381e93524252b7f0cc05a3d9455c-vp1.us.blockchain.ibm.com:444`

VP0

VP1

VP2

VP3

(feel free to change the peer url above)

[+ Network's Enroll IDs](#)

IBM Blockchain API

Interact with the enterprise blockchain through IBM Blockchain API

Block

[Show/Hide](#)

[List Operations](#)

[Expand Operations](#)

Blockchain

[Show/Hide](#)

[List Operations](#)

[Expand Operations](#)

Chaincode

[Show/Hide](#)

[List Operations](#)

[Expand Operations](#)

Network

[Show/Hide](#)

[List Operations](#)

[Expand Operations](#)

Registrar

[Show/Hide](#)

[List Operations](#)

[Expand Operations](#)

Transactions

[Show/Hide](#)

[List Operations](#)

[Expand Operations](#)

Welcome to the Starter Developer Network on IBM Blockchain!

IBM Blockchain API

Interact with the enterprise blockchain through IBM Blockchain API

Block

Show/Hide | List Operations | Expand Operations

GET

/chain/blocks/{Block}

Individual block information

Implementation Notes

The {Block} endpoint returns information about a specific block within the Blockchain. Note that the genesis block is block zero.

Response Class (Status 200)

[Model Details](#) | [Model Schema](#)

```
{
  "proposerID": "string",
  "timestamp": {
    "seconds": 0,
    "nanos": 0
  },
  "transactions": [
    {
      "type": "UNDEFINED",
      "chaincodeID": "string",
```

Response Content Type

application/json



Section 4

Where do we go from here

Using the Hello Chaincode tutorial

This tutorial guides you through using basic building blocks to code an elementary chaincode.

- What is chaincode?
- How do I implement chaincode?
- What dependencies exist?
- What are the major functions?
- How do I pass different values to my arguments?
- How do I securely enroll a user on my network?
- How do I compile my chaincode?
- How do I interact with my chaincode through the REST API?

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html#hellocc

Learn Chaincode

A tutorial to get you started with writing smart contracts for Hyperledger.

Deployment

In order to support multiple versions of the Hyperledger fabric, this repository uses branches in combination with gopkg.in URLs. What does this mean for beginners? Just pick the branch below and use the instructions for that branch to complete the tutorial

<https://github.com/IBM-Blockchain/learn-chaincode/blob/master/README.md>

Educate Yourself!

- Everything is available through the web:
 - <http://hyperledger.org>
 - <https://github.com/hyperledger>
 - <http://ibm.com/blockchain>

