

**IBM Cloud**



# **Analytics, Blockchain and Internet of Things (IoT)**

## **Instructor Lab Guide**





## Notices and Disclaimers

© Copyright IBM Corporation 2018.

The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. This information is based on current IBM product plans and strategy, which are subject to change by IBM without notice. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way.

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

IBM, the IBM logo and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at [ibm.com/legal/copytrade.shtml](http://ibm.com/legal/copytrade.shtml)

Other company, product and service names may be trademarks or service marks of others



## Document Revision History

Rev #	File Name	Date
1.0	Blockchain and IoT Instructor Lab Guide	9/14/2018

**Prepared & Revised by:**

Loren Murphy – [lmurphy@us.ibm.com](mailto:lmurphy@us.ibm.com)

Dave Wakeman – [dwakeman@us.ibm.com](mailto:dwakeman@us.ibm.com)

## Table of Contents

<b>Instructor Lab Guide.....</b>	<b>5</b>
<b>Lab Environment Overview .....</b>	<b>6</b>
<b>Setup Blockchain Environment .....</b>	<b>7</b>
<b>Clean Up Blockchain Environment .....</b>	<b>14</b>

## Instructor Lab Guide

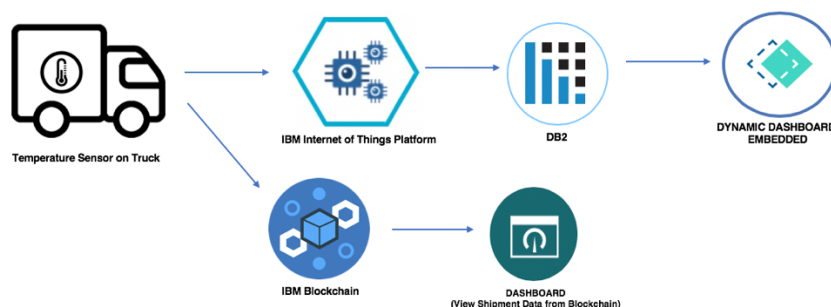
The purpose of this lab guide is to show you how to setup the workshop lab environment for the Blockchain and IoT Lab. The lab is based on the IBM Code Pattern, [“IoT Asset Tracking on a Blockchain”](#)

### Lab Overview

Our global economy and populations depend on safe delivery of perishable goods (food, medicine, livestock, etc.). Whenever public health officials issue a warning about bacterial outbreaks affecting the food supply, there are investigations into the source and cause of the contamination. Often these perishable goods are sensitive to environmental conditions during shipment. Were the perishable goods exposed to extreme temperatures? To preserve freshness, shipments of perishable goods might have refrigeration requirements because no one wants to eat unsafely warmed meat or bruised apples. If the shipment exceeds these temperature thresholds, the goods are likely damaged and might become a health hazard.

Tracking the conditions of the shipment across multiple participants using a blockchain provides verification and trust, while sensors within the shipment records environmental conditions in real-time. Is the truck refrigeration sufficient for this particular type of good? What temperature ranges were prescribed in the Smart Contract? Once it arrives at the final destination, is this shipment still safe or damaged?

This lab will simulate a temperature sensor within a shipment of perishable goods. As the truck travels to its final destination, the temperature, location, and reading time of shipment will be recorded in the blockchain. The shipment information will also be sent to the Watson IoT platform for further real-time analysis and persisted within Db2 on Cloud for further analysis and visualization.



## Lab Environment Overview

### Software and Tools

Software	Link
GitHub	<a href="https://github.com/team-wolfpack">https://github.com/team-wolfpack</a>
IBM Cloud	<a href="https://www.ibm.com/cloud/">https://www.ibm.com/cloud/</a>

**Note:** We know Composer is no longer being invested in; however, the lab should still work.

#### Troubleshooting Tips:

- If you received a “no response from server” error, stop then restart your Blockchain API service.

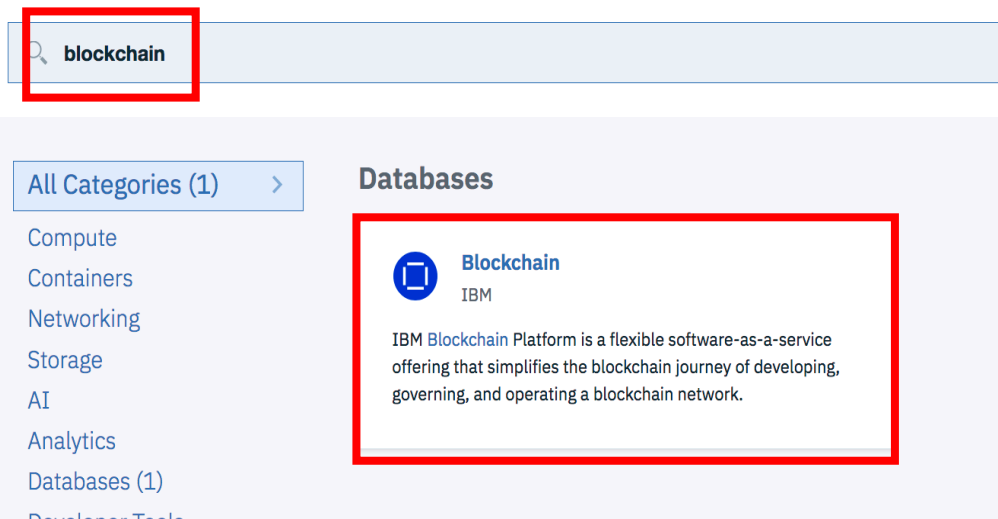
## Setup Blockchain Environment

### Action

#### 1.Setup Blockchain Environment

- Login to IBM Cloud account <https://console.bluemix.net/catalog/>
- In the catalog, search **Blockchain** and select the **Blockchain service**


#### Catalog



- Keep the default name or provide a unique name. Select the **Starter Membership Plan** and click **Create**

## Action

[← View all](#)



Blockchain

IBM

IBM Blockchain Platform is a flexible software-as-a-service offering that is delivered via the IBM Cloud. It enables network members to quickly get started developing and easily move to a collaborative environment. The platform simplifies your blockchain journey of developing, governing, and operating a network. Choose a membership plan based on your ecosystem needs.

[View Docs](#)
[Terms](#)

AUTHOR	IBM
PUBLISHED	09/07/2018
TYPE	Service
LOCATION	Sydney, Germany, United Kingdom, US South

Service name:

Blockchain-xu

Choose a region/location to deploy in:

US South

Choose an organization:

lrmurphy@us.ibm.com

Choose a space:

Boston

Features

- Use the IBM Blockchain Platform to simplify the developmental, governmental, and operational aspects of creating a blockchain solution. The following plans enable you to easily migrate from POC to pilot, all the way through to production on a secure, high performance, and fully scalable production network that you can't outgrow.
- For more information on developing, governing, and operating your blockchain network, see <https://ibm.biz/bcdocumentation>.
- Get \$500 towards your first network with Starter Plan, featuring an easy-to-use UI to reduce network administration and governance time, an iterative development platform and basic service levels for pilot evaluation or pre-production POCs.\* <http://ibm.biz/credits>
- Enterprise Plan offers a secure production environment and advanced service levels for production grade deployment, application development, and production testing.

Images

[Need Help?](#)  
[Contact IBM Cloud Support](#)

[Estimate Monthly Cost](#)  
[Cost Calculator](#)

Create

d. Go to the Starter Toolchain:

[https://console.bluemix.net/devops/setup/deploy/?repository=https%3A//github.com/IBM-Blockchain-Starter-Kit/blockchain-toolchain&branch=master&env\\_id=ibm%3Ayp%3Aus-south&deploy-region=ibm%3Ayp%3Aus-south&sampleRepo=https%3A//github.com/IBM-Blockchain-Starter-Kit/blockchain-sample-bootstrap](https://console.bluemix.net/devops/setup/deploy/?repository=https%3A//github.com/IBM-Blockchain-Starter-Kit/blockchain-toolchain&branch=master&env_id=ibm%3Ayp%3Aus-south&deploy-region=ibm%3Ayp%3Aus-south&sampleRepo=https%3A//github.com/IBM-Blockchain-Starter-Kit/blockchain-sample-bootstrap)



## Action



### Develop a Blockchain app with Hyperledger Composer

With this toolchain, you can develop and deploy a Blockchain application. By default, the toolchain uses a sample app, or you can link to your own GitHub repository. This toolchain is preconfigured for continuous delivery, source control, issue tracking, and online editing.

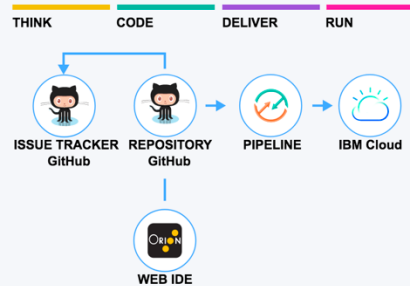
This toolchain uses tools that are part of the Continuous Delivery service. If an instance of that service isn't already in the selected organization, when you click **Create**, it is automatically added with the free [Lite](#) plan selected.

To get started, click **Create**.

For step-by-step instructions, follow the [tutorial](#).

#### TEMPLATE INFO

GIT URL <https://github.com/IBM-Blockchain-20180916130339005>  
 GIT BRANCH [master](#)



Toolchain Name:

blockchain-20180916130339005

Select Region:

US South

Select a resource group:

default

Create

- e. Click **Delivery Pipeline**. Change the blockchain service name to the **same name as your blockchain service** which you defined in step C. Rename or keep the default cloudant service name. Change the app name or keep default.

## Action

### Tool Integrations



GitHub



Eclipse Orion Web  
IDE



Delivery Pipeline

The Delivery Pipeline automates continuous deployment of your business network and client application.

**Blockchain service name:**

Blockchain-xu

**Cloudant NoSQL DB service name:**

Wallet-20180916130339005

**App name:**

blockchain-20180916130339005

f. Click **Create** to create a Cloud API Key. Click **Create**

IBM Cloud API Key:

IBM Cloud API Key

Create +

\*The value is required.

Note: Blockchain Starter Plan is only available in US South, United Kingdom and Sydney

Region:	Organization:	Space:
The IBM Cloud region	The Cloud Foundry organization	The Cloud Foundry space

\*The value is required.

\*The value is required.

\*The value is required.

### Create API key

The following API key will be created to deploy your application from the Delivery Pipeline.

API Key for blockchain-20180916130339005

API keys can be managed from **Manage > Security > Platform API keys**.

Cancel

Create

g. Select the same **Region**, **Organization** and **Space** for where you provisioned the blockchain service

## Action

The Delivery Pipeline automates continuous deployment of your business network and client application.

**Blockchain service name:** ⓘ

Blockchain-xu

**Cloudant NoSQL DB service name:** ⓘ

Wallet-20180916130339005

**App name:** ⓘ

blockchain-20180916130339005

**IBM Cloud API Key:** ⓘ

.....



Create +

**Note:** Blockchain Starter Plan is only available in US South, United Kingdom and Sydney

**Region:**

US South (Production) ▼

**Organization:**

NACHannels ▼

**Space:**

Wolfpack Demo Development ▼

h. Click **Create**

## Action

The Delivery Pipeline automates continuous deployment of your business network and client application.

**Blockchain service name:**  ⓘ

**Cloudant NoSQL DB service name:**  ⓘ

**App name:**  ⓘ

**IBM Cloud API Key:**  ⓘ 👁 Create +

**Note:** Blockchain Starter Plan is only available in US South, United Kingdom and Sydney

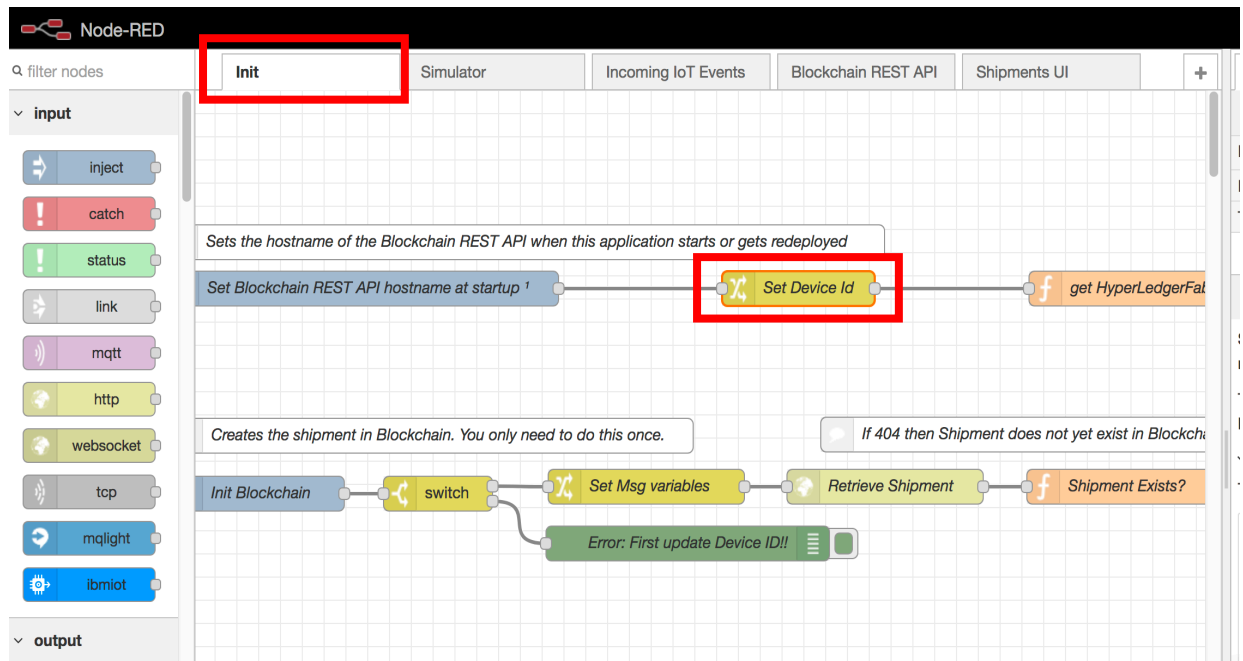
**Region:**  ▼ **Organization:**  ▼ **Space:**  ▼

Create

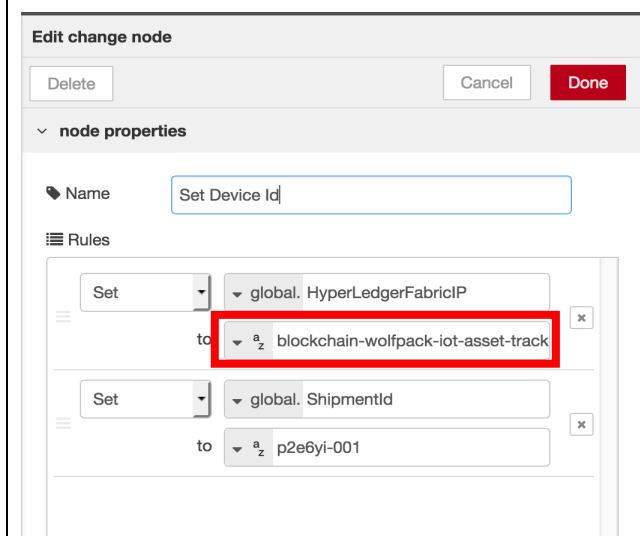
- i. Click on the “GitHub” link within the tool chain. Clone the repo locally to your computer.
- j. Unzip the IoT-Perishable-Network.zip file and move the IoT Perishable Network folder into the “Contracts” folder. **Delete the [readme.md file in the contracts folder](#) and also delete the [readme.md file in the apps folder](#)**
- k. Commit your changes and push them to GitHub. This will trigger the build step of the pipeline you just created.
- l. Once the build step finishes, the deploy step triggers. Once that finishes, go back to the dashboard. You should see a service was created. This is your composer rest server (API).
- m. Click on the service. And “Visit app URL.” If you want to pre-populate data for demos, go to “SetupDemo.” Click on “example value” to copy its contents into the value box. Delete the timestamp, and transactionID. And click “Try it out”.

## Action

- n. To make sure data was populated go to “Contract” , “get” and try the example value.
- o. In the Node-RED flow, make sure you change the **HyperLedgerFabricIP** to your Blockchain URL. This is found within the **Set Device ID** node in the **Init Flow**.



- p. **IMPORTANT:** Make sure you give the workshop participants the URL so they can also update their Set Device ID node.



The 'Edit change node' dialog box for the 'Set Device Id' node is shown. It includes a 'Delete' button, 'Cancel' and 'Done' buttons, and a 'node properties' section. The 'Name' field is 'Set Device Id'. The 'Rules' section contains two rules:

- Rule 1:** Set global. HyperLedgerFabricIP to blockchain-wolfpack-iot-asset-track. The value 'blockchain-wolfpack-iot-asset-track' is highlighted with a red box.
- Rule 2:** Set global. ShipmentId to p2e6yi-001.

## Clean Up Blockchain Environment

### Action

#### 1.Clean Up Blockchain Environment


- After the workshop, make sure you delete the participant's shipment IDs in the blockchain by following the steps below:
- Go to your IBM Cloud dashboard and click on your **Blockchain Application**

#### Cloud Foundry Applications

Name ^	Region	CF Org	CF Space	Memory (MB)	Status	
Bike Share Streaming-Demo	US South	NACHannels	NA ESA Demo	512	● Running (1/1)	⋮
Blockchain-wolfpack_iot-asset-tracker-netwo63e2600	US South	NACHannels	Wolfpack Demo ...	256	● Running (1/1)	⋮

- Select **Visit App URL** to open your Blockchain API Platform.

Cloud Foundry apps /


Blockchain-wolfpack\_iot-asset...
● Running
**Visit App URL**

Org: NACHannels Location: US South Space: Wolfpack Demo Development

#### IBM Blockchain Platform: Develop

AccelReading : A transaction named AccelReading	Show/Hide	List Operations	Expand Operations
Contract : An asset named Contract	Show/Hide	List Operations	Expand Operations
GpsReading : A transaction named GpsReading	Show/Hide	List Operations	Expand Operations
Grower : A participant named Grower	Show/Hide	List Operations	Expand Operations
Importer : A participant named Importer	Show/Hide	List Operations	Expand Operations
SetupDemo : A transaction named SetupDemo	Show/Hide	List Operations	Expand Operations
Shipment : An asset named Shipment	Show/Hide	List Operations	Expand Operations
ShipmentReceived : A transaction named ShipmentReceived	Show/Hide	List Operations	Expand Operations
Shipper : A participant named Shipper	Show/Hide	List Operations	Expand Operations
System : General business network methods	Show/Hide	List Operations	Expand Operations
TemperatureReading : A transaction named TemperatureReading	Show/Hide	List Operations	Expand Operations

- Select **Shipment** then **Get/Shipment**

## Action

**IBM Blockchain Platform: Develop**

AccelReading : A transaction named AccelReading

Contract : An asset named Contract

GpsReading : A transaction named GpsReading

Grower : A participant named Grower

Importer : A participant named Importer

SetupDemo : A transaction named SetupDemo

**Shipment : An asset named Shipment**

ShipmentReceived : A transaction named ShipmentReceived

Shipper : A participant named Shipper

System : General business network methods

TemperatureReading : A transaction named TemperatureReading

**IBM Blockchain Platform: Develop**

AccelReading : A transaction named AccelReading [Show/Hide](#) [List Operations](#) [Expand Operations](#)

Contract : An asset named Contract [Show/Hide](#) [List Operations](#) [Expand Operations](#)

GpsReading : A transaction named GpsReading [Show/Hide](#) [List Operations](#) [Expand Operations](#)

Grower : A participant named Grower [Show/Hide](#) [List Operations](#) [Expand Operations](#)

Importer : A participant named Importer [Show/Hide](#) [List Operations](#) [Expand Operations](#)

SetupDemo : A transaction named SetupDemo [Show/Hide](#) [List Operations](#) [Expand Operations](#)

**Shipment : An asset named Shipment** [Show/Hide](#) [List Operations](#) [Expand Operations](#)

**GET** /Shipment [Find all instances of the model matched by filter from the data source.](#)

**POST** /Shipment [Create a new instance of the model and persist it into the data source.](#)

**GET** /Shipment/{id} [Find a model instance by \(id\) from the data source.](#)

**HEAD** /Shipment/{id} [Check whether a model instance exists in the data source.](#)

**PUT** /Shipment/{id} [Replace attributes for a model instance and persist it into the data source.](#)

**DELETE** /Shipment/{id} [Delete a model instance by \(id\) from the data source.](#)

ShipmentReceived : A transaction named ShipmentReceived [Show/Hide](#) [List Operations](#) [Expand Operations](#)

Shipper : A participant named Shipper [Show/Hide](#) [List Operations](#) [Expand Operations](#)

System : General business network methods [Show/Hide](#) [List Operations](#) [Expand Operations](#)

TemperatureReading : A transaction named TemperatureReading [Show/Hide](#) [List Operations](#) [Expand Operations](#)

[ BASE URL: /api , API VERSION: 1.0.0 ]

- e. Click **Try it Out** to return a list of all the shipment IDs in the blockchain. Copy one of the Shipment IDs.

**Shipment : An asset named Shipment**
[Show/Hide](#)
[List Operations](#)
[Expand Operations](#)

**GET** /Shipment [Find all instances of the model matched by filter from the data source.](#)

Response Class (Status 200)  
Request was successful

Model [Example Value](#)

```

{
  "$class": "org.acme.shipping.perishable.Shipment",
  "shipmentId": "string",
  "type": "BANANAS",
  "status": "CREATED",
  "unitCount": 0,
  "contract": {},
  "temperatureReadings": [],
  "AccelReadings": [],
}
```

Response Content Type: application/json

Parameter	Value	Description
filter		Filter defining fields, when order, offset, and limit - m JSON-encoded string ({"something":"value"})

**Try it out!** [Hide Response](#)

**Response Body**

```

[
  {
    "$class": "org.acme.shipping.perishable.Shipment",
    "shipmentId": "01",
    "type": "MEDICINE",
    "status": "CREATED",
    "unitCount": 100,
    "contract": "resource:org.acme.shipping.perishable.Contract#CON_002",
    "temperatureReadings": [],
    "AccelReadings": [],
    "gpsReadings": []
  },
  {
    "$class": "org.acme.shipping.perishable.Shipment",
    "shipmentId": "02",
    "type": "MEDICINE",
    "status": "CREATED",
    "unitCount": 100,
    "contract": "resource:org.acme.shipping.perishable.Contract#CON_002",
    "temperatureReadings": [],
    "AccelReadings": [],
    "gpsReadings": []
  }
]
```

## Action

- f. Click **Delete /Shipment/{id}** and paste the Shipment ID you just copied into the **Value** box. Click **Try it Out**.

**Shipment : An asset named Shipment**

GET	/Shipment
POST	/Shipment
GET	/Shipment/{id}
HEAD	/Shipment/{id}
PUT	/Shipment/{id}
DELETE	/Shipment/{id}

DELETE /Shipment/{id}

Delete a model instance by {id} from the data source.

Response Class (Status 200)

Request was successful

Model Example Value

{}

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	01	Model id	path	string

Try it out!

- g. You should receive a **Response Body** of “no content” and a **Response Code** of “204”. You have now successfully deleted that Shipment ID within your blockchain. **Repeat steps D- F** for the remaining shipment IDs that were created within your workshop.

Response Body

no content

Response Code

204