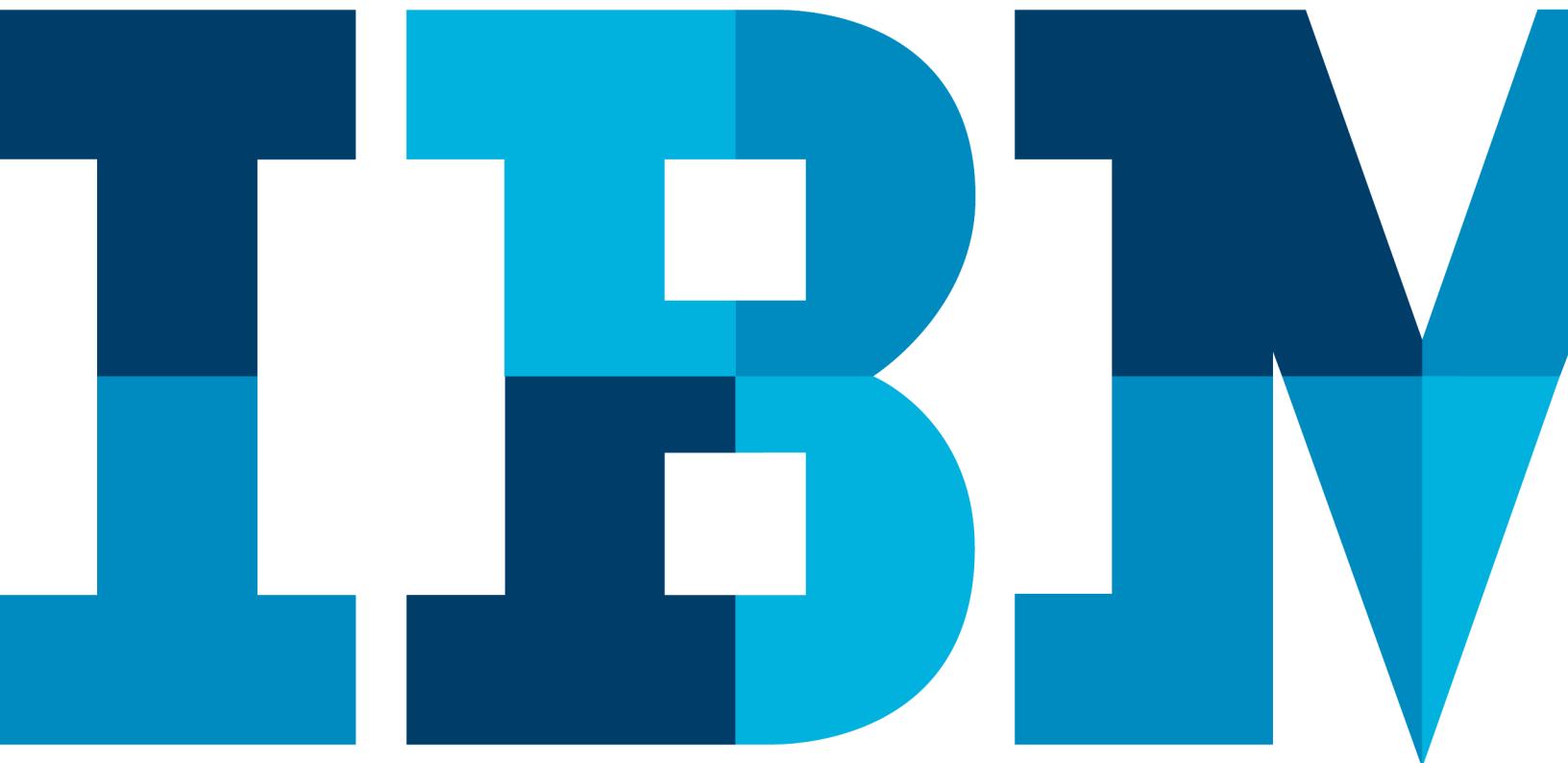


## IBM Blockchain Hyperledger Fabric Hands-On

Deploy and test your smart  
contract in IBM Blockchain  
Platform *Lab*



---

# 1 Contents

<b>1</b>	<b>Contents .....</b>	<b>2</b>
<b>2</b>	<b>Overview .....</b>	<b>3</b>
2.1	<i>Introduction.....</i>	3
2.2	<i>Prerequisites.....</i>	3
2.3	<i>IBM Blockchain platform – Starter plan initialisation .....</i>	3
<b>3</b>	<b>Discover the Starter Plan .....</b>	<b>6</b>
3.1	<i>Overview.....</i>	6
3.2	<i>Members.....</i>	7
3.3	<i>Channels.....</i>	8
3.4	<i>Notifications.....</i>	9
3.5	<i>Certificate Authority.....</i>	10
3.6	<i>APIs.....</i>	10
<b>4</b>	<b>Deploy your smartcontract .....</b>	<b>12</b>
4.1	<i>Install code .....</i>	12
<b>5</b>	<b>Manage your blockchain with Swagger API.....</b>	<b>21</b>
5.1	<i>Introduction to Swagger API.....</i>	21
5.2	<i>Get the status of your nodes.....</i>	23
5.3	<i>Get the list of CA users.....</i>	24
<b>6</b>	<b>Smartcontract execution.....</b>	<b>28</b>
6.1	<i>Preparation.....</i>	28
6.2	<i>Run the smartcontract .....</i>	29
6.3	<i>Check the transaction in the StarterPlan .....</i>	30
<b>Notices</b>	<b>31</b>	
<b>Appendix A. Trademarks and copyrights.....</b>		<b>33</b>

---

## 2 Overview

The aim of this lab is to experiment how to deploy and test a smartcontract in IBM Blockchain Platform (Starter Plan)

### 2.1 Introduction

*In the rest of the document, “bash terminal” “Linux terminal” or “terminal window” will refer to the terminal window described in the previous lab document.*

### 2.2 Prerequisites

You should have an IBM Cloud account to subscribe to and use the IBM Blockchain Platform Starter Membership Plan (referred to as IBM Blockchain Platform Starter Plan, or Starter Plan, in the rest of this document).

This lab requires

- a Smartcontract already developed in Golang for Hyperledger Fabric 1.x. (Refer to lab Lab HLF - SmartContract in Golang).  
If you have a SmartContract different from the one created during the lab “Lab HLF - SmartContract in Golang”, you can run this lab, but you will have to adapt it, depending on the transaction you will use.

NodeJS installed (refer to the Labhlf - preparation guide)

### 2.3 IBM Blockchain Platform – Starter Membership Plan initialization

The Starter Membership Plan is the IBM Blockchain Platform (Hyperledger Fabric) plan dedicated to development and Proof of Concept of Blockchain solutions.

Here, we are going to initiate a Blockchain Starter Membership Plan.

**PAY ATTENTION: This IBM Cloud service is charged for, but the fees are offered for the first month. After this first trial month, this service is only accessible if you have provided your credit card information in your IBM Cloud profile.**

**You will activate the service for the duration of the lab. Don't forget to remove, then before the end of the first month, you have to remove it.**

**Be aware that you can share one Starter Plan between several developers.**

Access to your IBM Cloud dashboard (<https://console.bluemix.net/catalog>) and search for “blockchain” in the search field as shown in the following picture:

IBM Cloud Catalog Docs Support Manage

Search for resource...

Menu Catalog

blockchain

All Categories (1) &gt;

**Databases****Blockchain**  
IBM

IBM Blockchain Platform is a flexible software-as-a-service offering that simplifies the blockchain journey of developing, governing, and operating a blockchain network.

Click on Blockchain frame (under Databases).

In the following panel, you can leave the default information. At the bottom of the page, you will be proposed 2 options:

- The Starter Membership Plan.
- The Enterprise Membership Plan.

**Select the Starter Membership Plan.**

IBM Cloud Catalog Docs Support Manage

Search for resource...

Olivier VALLOD's Account

[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-bv-starter

**Choose a region/location to deploy in:** United Kingdom

**Choose an organization:** olivier.vallod@fr.ibm.com

**Choose a space:** devUK

**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**

Images can be screen captures, slides, or videos. Click an image to view the details.

Pricing Plans

Monthly prices shown are for country or region: France

PLAN	FEATURES	PRICING
<input checked="" type="checkbox"/> Starter Membership Plan	<p>Get \$500 towards your first Starter Plan network - sign up today  <a href="#">Hyperledger Business Network Archive</a></p> <p>This plan entitles a member of the Blockchain Platform to:</p> <ul style="list-style-type: none"> <li>- With a single click, setup and a fully functional kick starter network that is configured with 2 organizations and 1 peer per organization.</li> <li>- This easily deployed development environment, includes all components of the blockchain network, quickly install and instantiate chaincode or import your business network archive (.BNA).</li> <li>- Also included are sample applications and links to 'how to' instructional videos and documentation.</li> <li>- Default Capabilities, along with all the UI capabilities provided in the Enterprise plan:</li> <li>- Ability to import a .BNA file that was created from Hyperledger Composer, or your native chaincode.</li> <li>- Simulate a blockchain network with multiple organizations under one account</li> <li>- Exposure of the APIs via Swagger.</li> </ul> <p>The IBM Blockchain Platform is the only fully integrated blockchain platform designed to accelerate the development, governing, and operation of a multi-institution business network. Starter Membership Plan has been created to help easily deploy sample applications, test applications, and begin to grow your network. It is not suitable for production workloads though. Starter Membership Plan includes Hyperledger Composer integration and an easy on-ramp to recognize the value of a production network deployed on the Enterprise Membership Plan. "Credits can cover the first month's fees for two peers and one membership fee. Please wait until the credits are in your account before accessing Starter Plan to avoid initial charges.</p>	<p>€188.00 EUR/Membership Fee per Month          €94.03 EUR/Small Peer</p>
<input type="checkbox"/> Enterprise Membership Plan	<p>This plan entitles a member of the Blockchain Platform to:</p> <p>All Starter Plan features, combined with premium support options, a secure blockchain environment for early production workloads, and added layers of security.</p> <p>Additional Capabilities:</p> <ul style="list-style-type: none"> <li>- Fault Tolerance that provides High Availability for ordering and certificate authority.</li> <li>- Secure Service Container that protects a shared compute with isolated runtime and data.</li> <li>- Hardware Security Module.</li> </ul>	<p>€752.00 EUR/Membership Fee          €752.00 EUR/Small Peer</p>

Click on the Create button. Then it displays the following windows (it may take several minutes), describing the Hyperledger fabric environment created. Click on the Launch button.

Manage

Service credentials

Connections

Dashboard /

 Blockchain-ov-starter

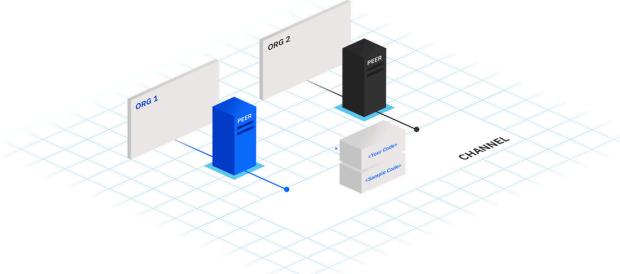
Location: United Kingdom    Org: olivier.vallod@fr.ibm.com    Space: devUK

**Network created!**

Your Starter Plan network has two organizations, each with its own peer.

With this network you can test chaincode, deploy samples, and invite other members to collaborate.

**Launch**



## 3 Discover the Starter Plan

In this section, you will navigate between the different screens of the starter plan in order to discover the different available features.

### 3.1 Overview

This provides the list of Blockchain resources accessible by this organization.

At the top right of the window, you can see which organization you are operating “Company A (org1)”. We observe there is one Orderer, one CA (Certificate Authority) and one Peer.

The screenshot shows the 'Overview' section of the IBM Blockchain Starter Plan. The left sidebar lists 'Everest Network - VZ6' under 'MY NETWORK' and includes links for 'Overview', 'Members', 'Channels', 'Notifications', 'Certificate Authority', 'APIs', 'MY CODE', 'Develop code', and 'Install code'. The main content area has a title 'Overview' with a sub-instruction: 'View and manage network resources for your organization. You can stop or start your resources and view the log file of the resource by selecting View Logs under "Actions". [Learn more](#)'. Below this is a table with three rows, each representing a resource: 'Orderer' (name 'orderer', status 'Running'), 'CA' (name 'org1-ca', status 'Running'), and 'Peer' (name 'org1-peer1', status 'Running'). Each row has a 'More Actions' button represented by three dots. At the bottom of the table, there are buttons for 'Remote Peer Configuration', 'Connection Profile', and 'Add Peers'.

Type	Name	Status	Actions
Orderer	orderer	Running	⋮
CA	org1-ca	Running	▶ ⏪ ⋮
Peer	org1-peer1	Running	▶ ⏪ ⋮

If you click on the top right corner, you will get a menu to switch organizations. Under SWITCH ORGANIZATION select Company B, then you will get the list of resources for Company B

IBM Blockchain Starter Plan

Cookie Preferences

OLIVIER VALLOD ^  
Operating as: Company A (org1)

Everest Network - VZ6

MY NETWORK

Overview Members Channels Notifications Certificate Authority APIs MY CODE Develop code Install code

Overview Storage

Learn more

Remote Peer Configuration Connection Profile Add Peers

TYPE	NAME	STATUS	ACTIONS
Orderer	orderer	Running	⋮
CA	org1-ca	Running	↻ ⏪ ⋮
Peer	org1-peer1	Running	↻ ⏪ ⋮

Go back to Company A.

## 3.2 Members

In the Members view, you can see the members part of your Business Network are.

IBM Blockchain Starter Plan

Cookie Preferences

OLIVIER VALLOD ^  
Operating as: Company A (org1)

Everest Network - VZ6

MY NETWORK

Overview Members Channels Notifications Certificate Authority APIs MY CODE Develop code Install code

Members Certificates

Add Member

MEMBERS (2/15)	MSP ID	REQUESTER	STATUS	ACTION
Company A olivier.vallod@fr.ibm.com	org1	--	Joined	✖
Company B olivier.vallod@fr.ibm.com	org2	--	Joined	✖

You can rename the member: rename Company A to Supplier, Company B to Carrier.

The Add Member button allows to either directly create a new member in this Business Network, or send a request by email (with instructions) to create a new member which will join this Business Network.

The tab Certificates displays the attached client certificates.

### 3.3 Channels

The channel view displays the channels in which this organization is participating.

The screenshot shows the IBM Blockchain Starter Plan interface. The top navigation bar includes 'IBM Blockchain Starter Plan', 'Cookie Preferences', and a user profile 'OLIVIER VALLOD' operating as 'Company A (org1)'. On the left, a sidebar titled 'Everest Network - VZ6' lists 'MY NETWORK' sections: Overview, Members, **Channels** (selected), Notifications, Certificate Authority, APIs, and 'MY CODE' sections: Develop code, Install code. The main content area is titled 'Channels' and contains a message: 'You must have a channel to propose a transaction. If you're not a member of any channels, you can create one by clicking "Request Channel". [Learn more](#)'. Below this is a search bar 'Search Channels' and a blue button 'Request Channel'. A table lists channels with columns: ID, TIME CREATED, BLOCK HEIGHT, PEERS, and ACTIONS. One row is shown: 'defaultchannel' created on '10/12/18 06:54 GMT+0200' at 'BLOCK HEIGHT 3' with 'PEERS org1-peer1' and an 'Actions' menu icon. At the bottom, a note says: '\* Channels created with the SDK cannot be edited here (yet)'.

When clicking on the line where the channel is listed, you access to the view of the activity and configuration of this channel.

The first tab (Channel overview) provides the statistics on the channel activity (number of blocks, instantiations...) as well as the list of the blocks.

The detail of a block can be seen clicking on the block line, then clicking on the menu icon in the column actions.

The screenshot shows the IBM Blockchain Starter Plan interface. On the left, a sidebar menu includes: Overview, Members, **Channels**, Notifications, Certificate Authority, APIs, MY CODE, Develop code, Install code, Try samples, and Get help. The Channels item is currently selected. In the main content area, the title is "defaultchannel" with the subtitle "View channel activity and configuration". Below this, there are four summary cards: "TOTAL BLOCKS" (3), "TIME SINCE LAST TRANSACTION" (19 mins), "RECENT INSTANTIATIONS" (0), and "RECENT INVOCATIONS" (2). A table titled "Channel Overview" lists three transactions with columns: TIME, BLOCK NUMBER, INSTANTIATIONS, and INVOCATIONS. The first transaction is highlighted.

TIME	BLOCK NUMBER	INSTANTIATIONS	INVOCATIONS
> 10/12/18 06:54 GMT+0200	2	0	1
> 10/12/18 06:54 GMT+0200	1	0	1
> 10/12/18 06:54 GMT+0200	0	0	0

The second tab (Members) displays the members participating to this channel and their permissions. These permissions can be changed, from the channel list, using the action “Edit channel”.

The screenshot shows the IBM Blockchain Starter Plan interface. The sidebar menu is identical to the previous screenshot. The Members tab is currently selected. The main content area shows the title "defaultchannel" and subtitle "View channel activity and configuration". Below this, there are four tabs: Channel Overview, **Members**, Access Control List, and Chaincode. The Members tab is active. A table lists members with columns: NAME, EMAIL, and PERMISSIONS. Two entries are shown: Company A (olivier.vallod@fr.ibm.com) with permissions Operator, Writer, Reader, and Company B (olivier.vallod@fr.ibm.com) with the same permissions.

NAME	EMAIL	PERMISSIONS
Company A	<a href="#">olivier.vallod@fr.ibm.com</a>	Operator Writer Reader
Company B	<a href="#">olivier.vallod@fr.ibm.com</a>	Operator Writer Reader

The Access Control List and Chaincode tabs, will be seen later.

### 3.4 Notifications

The Notifications view displays the notifications concerning the organization. These notifications may require an action from the organization (for example, approve the addition of a channel). The Pending tab displays the notifications with pending actions.

The screenshot shows the 'Notifications' section of the IBM Blockchain Starter Plan. The left sidebar has a dark theme with white text. The 'Notifications' item is highlighted with a blue background. The main content area has a light gray header with the title 'Notifications'. Below it, a message says: 'You get a notification whenever a creation or update request for a channel that you are included is submitted. Review, vote, and submit channel requests with the buttons under "Actions".' There are three tabs at the top: 'All (1)', 'Pending (0)', and 'Completed (1)'. The 'All (1)' tab is selected. A search bar with the placeholder 'Search notifications' is below the tabs. A table follows, with columns: NAME, DATE UPDATED, MY STATUS, and ACTION. One row is shown: 'Channel Request Join "defaultchannel"' by 'undefined' on '12 October, 2018 - 6:54:11 AM'. The status is 'Vote Accepted' with a green dot icon. The action button is labeled 'Review Request'.

## 3.5 Certificate Authority

This view displays the certificates of the solution.

The screenshot shows the 'Certificate Authority' section of the IBM Blockchain Starter Plan. The left sidebar has a dark theme with white text. The 'Certificate Authority' item is highlighted with a blue background. The main content area has a light gray header with the title 'Certificate Authority'. Below it, a message says: 'View and manage certificate authority of user identities and client applications. [Learn more](#)'. A blue button labeled '+ Add User' is on the right. A table below lists users with columns: ID, TYPE, and AFFILIATION. Two rows are shown: 'admin' (Client) and 'peer1' (Client, affiliated with 'org1'). A blue button labeled '+ Generate Certificate' is next to the 'peer1' row.

## 3.6 APIs

This view provides access to APIs to manage the blockchain environment. We will use this APIs later in the lab.

IBM Blockchain Starter Plan

Cookie Preferences

OLIVIER VALLOD ▾  
Operating as: Company A (org1)

Overview

Members

Channels

Notifications

Certificate Authority

**APIs**

MY CODE

Develop code

Install code

Try samples

Get help

## APIs

Interact with the network by using APIs in the Swagger UI. You can also use the network credentials and integrate the APIs to your own application. [Learn more](#)



**API reference list**  
Use the Swagger UI to try out the available catalog of REST APIs against the network.  
[Swagger UI](#)



**Network credentials**  
Use these network credentials to access the resource endpoints that your application needs to. Use "key" as your "Username" and "secret" as your "Password" when you authorize your APIs in the Swagger UI.

```
{  
  "url": "https://blockchain-starter.eu-gb.bluemix.net",  
  "network_id": "n49d325bbb1ab4866827b494b23c2e24e",  
  "key": "org1",  
  "secret": "XXXXXXXXXX"  
}
```

Show secret

## 4 Deploy your smartcontract

Now you will learn how to deploy and instantiate your SmartContract.  
Go to the “Install code” page.

### 4.1 Install code

Go to the view Install code. Then Click on Choose peer: select org1-peer1

The screenshot shows the IBM Blockchain Starter Plan interface. On the left, there is a sidebar with the following menu items:

- Overview
- Members
- Channels
- Notifications
- Certificate Authority
- APIs
- MY CODE
- Develop code
- Install code** (this item is highlighted with a blue background)
- Try samples
- Get help

The main content area is titled "Install code". It contains a sub-instruction: "Chaincode must be installed on a peer. Select a peer and then install a chaincode on it. After you install the chaincode, you can request to instantiate it on a channel by clicking the Instantiate button under "Actions" for that chaincode. Clicking elsewhere on the chaincode will show you what channels the chaincode is instantiated on. [Learn more](#). Do you already have a .bna file to deploy? See our [deployment guide](#)."

Below this, there is a dropdown menu labeled "Choose peer..." and a blue button labeled "Install Chaincode".

A table is present with the following columns: CODE ID, VERSION, and ACTIONS. The table body is currently empty, displaying the message "No installed chaincodes found."

Click on the button “Install Chaincode” then fill in the info and submit:

Name: labhf-<initials>-sc

Version: v0.0.1

Chaincode Type: Golang

The screenshot shows the IBM Blockchain Starter Plan interface. On the left sidebar, under 'MY CODE', 'Install code' is selected. The main area is titled 'Install code' and 'Install chaincode on org1-peer1'. It shows a file selection dialog with 'labhf-ov-sc' selected, version 'v0.0.1', and 'Choose files' button. Chaincode Type is set to 'Golang'. A note says 'I can request to instantiate it on a channel by clicking the Instantiate button under "Actions" for that chaincode. Clicking elsewhere on the chaincode will show you what channels the chaincode is instantiated on. [Learn more](#). Do you already have a .bna file to deploy? See our [deployment guide](#)'. There is a 'Cancel' and a 'Submit' button.

The smartcontract is installed on the peer of the Company A.

The screenshot shows the IBM Blockchain Starter Plan interface. On the left sidebar, under 'MY CODE', 'Install code' is selected. The main area is titled 'Install code' and displays a table of installed chaincodes. One entry is shown: 'labhf-ov-sc' with version 'v0.0.1'. There is a 'Cookie Preferences' button at the top right and a user info bar 'OLIVIER VALLOD Operating as: Company A (org1)'. A note at the top of the main area says 'Chaincode must be installed on a peer. Select a peer and then install a chaincode on it. After you install the chaincode, you can request to instantiate it on a channel by clicking the Instantiate button under "Actions" for that chaincode. Clicking elsewhere on the chaincode will show you what channels the chaincode is instantiated on. [Learn more](#). Do you already have a .bna file to deploy? See our [deployment guide](#)'.

Now we are going to install it on the peer of Company B. Switch to Company B (click on your user name at the top right, then select Company B instead of Company A under “SWITCH ORGANIZATION”)

Click here to switch between the organizations you control.

**OLIVIER VALLOD** Operating as: Company B (org2)

**NETWORK PREFERENCES**

**SWITCH ORGANIZATION**

Company B (org2)

Getting started  
Reset Network  
Logout

**Overview** Storage

**Remote Peer Configuration** **Connection Profile** **Add Peers**

Type	Name	Status	Actions
Orderer	orderer	Running	⋮
CA	org2-ca	Running	↻ ⚡ ⋮
Peer	org2-peer1	Running	↻ ⚡ ⋮

Do the same to install the smartcontract as you just did for org1-peer1.

IBBlockchain Starter Plan

Cookie Preferences

**OLIVIER VALLOD** Operating as: Company B (org2)

Overview Members Channels Notifications Certificate Authority APIs MY CODE Develop code Install code

**Install code**

Chaincode must be installed on a peer. Select a peer and then install a chaincode on it. After you install the chaincode, you can request to instantiate it on a channel by clicking the Instantiate button under "Actions" for that chaincode. Clicking elsewhere on the chaincode will show you what channels the chaincode is instantiated on. [Learn more](#). Do you already have a .bna file to deploy? See our [deployment guide](#).

org2-peer1

**Install Chaincode**

Code ID	Version	Actions

No installed chaincodes found.

The screenshot shows the 'Install code' section of the IBM Blockchain Starter Plan. On the left sidebar, under 'MY CODE', 'Install code' is selected. The main area displays a form titled 'Install chaincode on org2-peer1'. It includes fields for 'Chaincode ID' (labhlf-ov-sc), 'Version' (v0.0.1), 'Choose files' (labhlf-ov-sc.go), 'Chaincode Type' (Golang), and 'Actions' (Install Chaincode, Cancel, Submit). A tooltip indicates that requesting chaincode will show the deployment guide.

Once the smartcontract is installed on both peers, go back to Company A, to instantiate the smartcontract.

The screenshot shows the list of installed chaincodes. Under 'MY CODE', 'Install code' is selected. The main area displays a table with columns 'CODE ID', 'VERSION', and 'ACTIONS'. One entry is shown: 'labhlf-ov-sc' with version 'v0.0.1'. The 'Actions' column for this row contains a three-dot menu icon.

You can see the smartcontract in the list of chaincode. Click on the Actions button on the line of your smartcontract and select Instantiate.

The screenshot shows the 'Install code' section of the IBM Blockchain Starter Plan. On the left sidebar, 'Install code' is selected. The main area has a heading 'Install code' with a sub-instruction: 'Chaincode must be installed on a peer. Select a peer and then install a chaincode on it. After you install the chaincode, you can request to instantiate it on a channel by clicking the Instantiate button under "Actions" for that chaincode. Clicking elsewhere on the chaincode will show you what channels the chaincode is instantiated on. [Learn more](#). Do you already have a .bna file to deploy? See our [deployment guide](#).' Below this is a dropdown menu 'Choose peer...' and a blue button 'Install Chaincode'. A table lists chaincodes with columns 'CODE ID', 'VERSION', and 'ACTIONS'. One row is shown: 'labhf-ov-sc' (v0.0.1). The 'ACTIONS' column for this row contains a dropdown menu with options: 'Instantiate', 'Update', and 'View Channels...'. The 'Instantiate' option is highlighted.

In the “Instantiate Chaincode” panel, leave the fields with the default value. We didn’t specify any argument in the init function so no need to fill in the Arguments field.  
Click Next

The screenshot shows the 'Instantiate chaincode' panel. The left sidebar has 'Install code' selected. The main area has a heading 'Instantiate chaincode' with a note: 'Enter the arguments for the 'init()' function as a comma separated list.' Below this is a text input field 'Arguments:' containing 'ex: a, 200, b, 250'. To the right is a blue button 'Install Chaincode'. Further down are fields for 'Channel:' (set to 'defaultchannel') and 'Chaincode Type:' (set to 'Golang'). At the bottom are three buttons: 'Reset', 'Cancel', and 'Next' (highlighted in blue).

Keep Simple Policy selected for the endorsement policy type.

The screenshot shows the 'IBM Blockchain Starter Plan' interface. On the left sidebar, under 'MY CODE', the 'Install code' option is selected. The main panel displays a step titled 'Instantiate chaincode' with the sub-step 'Select an endorsement policy type.' A radio button is selected for 'Simple Policy', which is described as 'Select the channel members that need to endorse transactions.' Below this, there is a note: 'Please select the appropriate network members needed to participate in the new policy.' A dropdown menu shows 'All Available Members Have Been Added'. At the bottom right of the panel are 'Cancel', 'Previous', and 'Submit' buttons.

## Submit the Instantiation.

Now we are going to check the instantiation in the list of blocks. The instantiation is subject to a transaction which is stored in the chain.

Go to the Channels view.

The screenshot shows the 'IBM Blockchain Starter Plan' interface. On the left sidebar, the 'Channels' option is selected. The main panel displays a section titled 'Channels' with a note: 'You must have a channel to propose a transaction. If you're not a member of any channels, you can create one by clicking "Request Channel". [Learn more](#)'.

Below this, there is a search bar labeled 'Search Channels' and a blue button labeled '+ Request Channel'.

ID	TIME CREATED	BLOCK HEIGHT	PEERS	ACTIONS
defaultchannel	10/12/18 07:23 GMT+0200	4	org1-peer1	⋮

\* Channels created with the SDK cannot be edited here (yet)

Click on the channel “defaultchannel” to open it and access its statistics.

**defaultchannel**

View channel activity and configuration

TIME	BLOCK NUMBER	INSTANTIATIONS	INVOCATIONS
> 10/12/18 07:33 GMT+0200	3	1	0
> 10/12/18 07:23 GMT+0200	2	0	1
> 10/12/18 07:23 GMT+0200	1	0	1

You observe that there is a block where the Instantiations counter is 1. Open the detail of this block.

**defaultchannel**

View channel activity and configuration

TIME	BLOCK NUMBER	INSTANTIATIONS	INVOCATIONS
> 10/12/18 07:33 GMT+0200	3	1	0

DATE	TYPE	UUID	CHAINCODE ID(S)	ACTIONS
10/12/18 07:33 GMT+0200	INstantiate	b6b088ff4c9f0bdd21804498 3a1ca02d747e036f45e67c09 954811fef51eb53		⋮

Now, let see the log of the Peer. Go the “Overview” view, then click on the Actions button of the line of the org1-peer1, and select View logs.

The screenshot shows the IBM Blockchain Starter Plan interface. On the left sidebar, there are several sections: Overview, Members, Channels, Notifications, Certificate Authority, APIs, MY CODE (with options to Develop code, Install code, Try samples, and Get help), and a bottom section for Help. The main content area is titled "Overview" and contains a sub-section "Storage". It lists three network resources: "Orderer" (NAME: "orderer", STATUS: Running), "CA" (NAME: "org1-ca", STATUS: Running), and "Peer" (NAME: "org1-peer1", STATUS: Running). Each resource has a "View Logs" button and a "Remove" button. At the top right, there are buttons for "Remote Peer Configuration", "Connection Profile", and "+ Add Peers". The top right corner shows the user's name "OLIVIER VALLOD" and the company "Operating as: Company A (org1)".

Observe the different info regarding the instantiation.

The screenshot shows a Kibana dashboard with the title "IBM Cloud Logging" and "Kibana 5 Admin". The top right shows the user's email "olivier.vallod@fr.ibm.com" and account information "Olivier VALLOD's Account | olivier.vallod@fr.ibm.com | devUK". The left sidebar has tabs for Discover, Visualize, Dashboard, and Management, with "Discover" selected. The main area shows a log search result for the filter "[logstash-4a958db0-522d...". The search results table has columns for "@timestamp" and "message". The table shows multiple log entries from October 12th, 2018, at various times, detailing the instantiation process of the peer. The bottom right of the dashboard shows a bar chart with the y-axis "Count" ranging from 0 to 100 and the x-axis "Time" showing dates from September 15 to October 9. The chart has a single blue bar reaching the 100 mark.

You can save this filter (org1-peer1): click on save, then fill in the save search field with org1-peer1

## IBM Blockchain

IBM Cloud Logging   Kibana 5   Admin   olivier.vallod@fr.ibm.com  
Olivier VALLOD's Account | olivier.vallod@fr.ibm.com | devUK

**kibana** 93 hits

Save Search   New   Save   Open   30 seconds   Last 30 days

Discover   Visualize   Dashboard   Management

New Saved Search   Save

NETWORK\_ID\_str:"nde911e8ac6e24386840db89fcb32438f" && NODE\_NAME\_str:"org1-peer1"

[logstash-4a958db0-522d-...]

Selected Fields: t message

Available Fields: @timestamp, @version, ALCH\_ACCOUNT\_ID\_str, ALCH\_TENANT\_ID, NETWORK\_ID\_str, NODE\_NAME\_str, NODE\_TYPE\_str, \_id, \_index, \_score

Count   Time   message

September 12th 2018, 09:23:07.334 - October 12th 2018, 09:23:07.334 — by 12 hours

2018-09-15 02:00   2018-09-19 02:00   2018-09-23 02:00   2018-09-27 02:00   2018-10-01 02:00   2018-10-05 02:00   2018-10-09 02:00   @timestamp per 12 hours

▶ October 12th 2018, 07:34:10.666 2018-10-12 05:33:59.043 UTC [cceventmgmt] HandleStateUpdates -> INFO 052[0m Channel [defaultchannel]: Handling LSCC state update for chaincode [labhf-ov-sc]

▶ October 12th 2018, 07:34:10.666 2018-10-12 05:33:59.035 UTC [committer/txvalidator] Validate -> INFO 051[0m [defaultchannel] Validated block [3] in 15ms

▶ October 12th 2018, 07:34:10.666 2018-10-12 05:33:59.019 UTC [msp] DeserializeIdentity -> INFO 04f[0m Obtaining identity

To facilitate the navigation between the different logs, you can do the same for the other components:  
In the filter criteria, <NETWORK\_ID\_str:"nde911e8ac6e24386840db89fcb32438f" &&  
NODE\_NAME\_str:"org1-peer1"> replace the value of NODE\_NAME\_str successively by

- org2-peer1
- orderer
- org1-ca
- org2-ca

IBM Cloud Logging   Kibana 5   Admin   olivier.vallod@fr.ibm.com  
Olivier VALLOD's Account | olivier.vallod@fr.ibm.com | devUK

**kibana** Org1-peer1 93 hits

New   Save   Open   30 seconds   Last 30 days

Discover   Visualize   Dashboard   Management

Open Search   Saved Searches Filter...   Name ▲   5 of 5   Manage Saved Searches

Orderer

Org1-ca

Org1-peer1

Org2-ca

Org2-peer1

NETWORK\_ID\_str:"nde911e8ac6e24386840db89fcb32438f" && NODE\_NAME\_str:"org1-peer1"

[logstash-4a958db0-522d-...]

Selected Fields: t message

Available Fields: @timestamp

Count   Time   message

September 12th 2018, 09:30:28.331 - October 12th 2018, 09:30:28.331 — by 12 hours

2018-09-15 02:00   2018-09-19 02:00   2018-09-23 02:00   2018-09-27 02:00   2018-10-01 02:00   2018-10-05 02:00   2018-10-09 02:00   @timestamp per 12 hours

## 5 Manage your blockchain with Swagger API

### 5.1 Introduction to Swagger API

Go to the APIs view.

The screenshot shows the IBM Blockchain Starter Plan interface. On the left, there's a sidebar with options like Overview, Members, Channels, Notifications, Certificate Authority, and APIs (which is selected). Below that are sections for MY CODE (Develop code, Install code, Try samples) and Get help. The main content area is titled "APIs" and contains a sub-section "API reference list" with a note about using the Swagger UI to try out the catalog of REST APIs against the network. It also features two circular icons: one with three horizontal lines and another with a lock symbol. To the right, there's a "Network credentials" section with a JSON snippet showing network\_id, key, and secret values, and a "Show secret" link at the bottom right.

This page provides the Network credentials required to run the API.

Click on “Show secret” link at the bottom right of the page, then save the informations network\_id, key and secret. You will have to use them in the API in the next chapters.

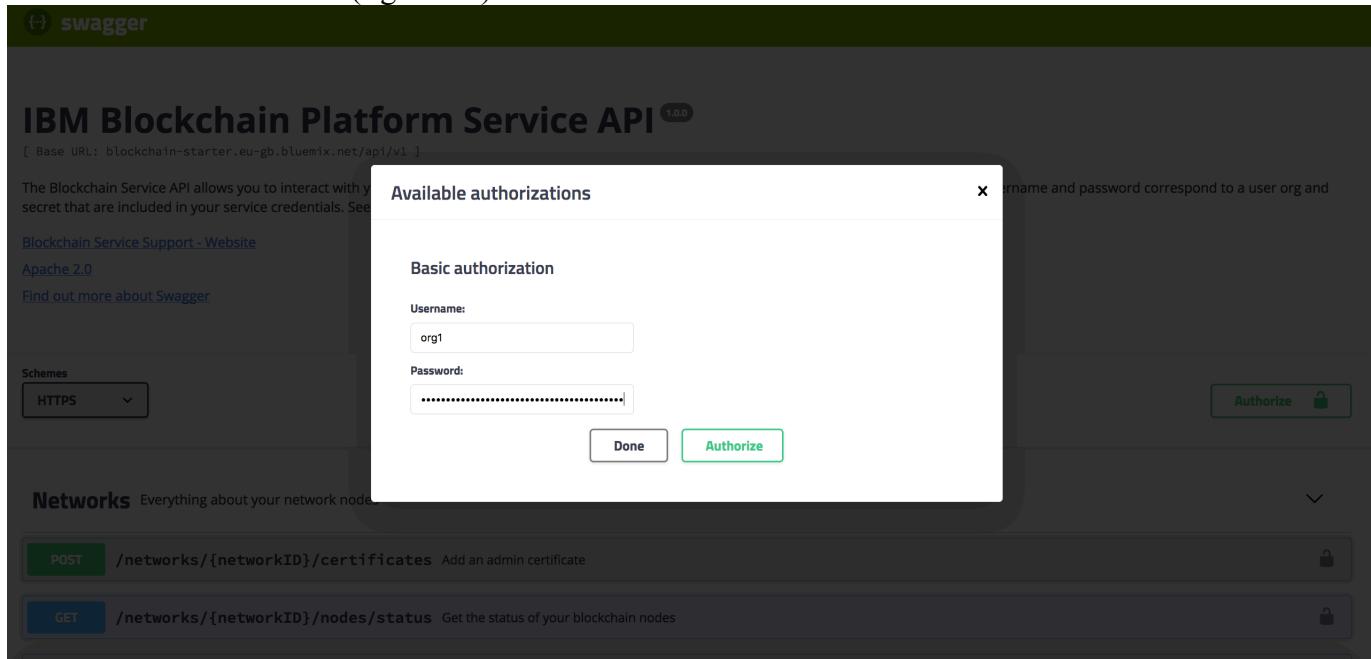
Click on the “Swagger UI” link.

The screenshot shows the IBM Blockchain Platform Service API Swagger UI. At the top, it says "IBM Blockchain Platform Service API 1.0.0" and provides the base URL: "blockchain-starter.eu-gb.bluemix.net/api/v1". Below this, there's a note about the API being authenticated using basic auth with username and password corresponding to a user org and secret. Links for "Blockchain Service Support - Website", "Apache 2.0", and "Find out more about Swagger" are available. The interface includes a "Schemes" dropdown set to "HTTPS" and an "Authorize" button. A "Networks" section is expanded, showing two API endpoints: a POST method for adding certificates to a network node and a GET method for getting the status of blockchain nodes. Both endpoints have their respective URLs and descriptions listed.

## IBM Blockchain

First before using the APIs, you must be authorized. So, you must provide key and secret that you have previously saved.

Click on Authorize button (right side)

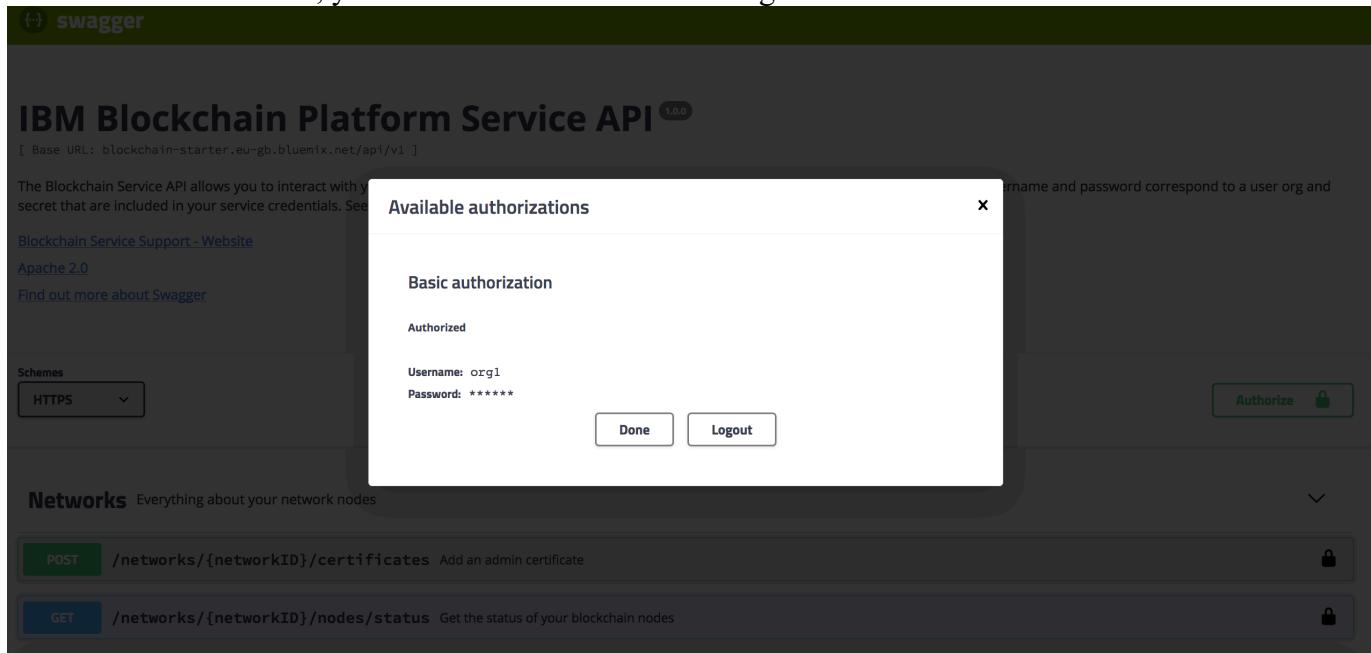


The screenshot shows the IBM Blockchain Platform Service API documentation. A modal window titled "Available authorizations" is open. It contains a "Basic authorization" section with "Username" set to "org1" and "Password" redacted. There are "Done" and "Authorize" buttons. The background shows sections for "Networks" and "Schemas".

The Username is the key you retrieved in the previous panel the Password is the secret you previously saved.

Once filled in, click on Authorize.

You are now authorized, you can close the window clicking on Done.



The screenshot shows the same API documentation with the modal window still open. The "Basic authorization" section now shows "Authorized", "Username: org1", and "Password: \*\*\*\*\*". The "Done" and "Logout" buttons are visible. The background sections for "Networks" and "Schemas" are also present.

Let's start using APIs to handle the Hyperledger Fabric environment.

## 5.2 Get the status of your nodes

Click on the node/status API. Then click on “Try it out” button.

**GET** /networks/{networkID}/nodes/status Get the status of your blockchain nodes

Returns the operational status of all your nodes on the network

**Parameters**

Name	Description
<b>networkID</b> * required	ID of the IBP network string (path)

**Responses**

Code	Description
200	<b>OK</b>

Example Value | Model

```
{
  "exampleNodeID": {
    "status": "running",
    "node_type": "fabric-ca"
  }
}
```

Response content type application/json

Fill in the networkID (retrieved on the [APIs view of the Blockchain Monitor](#)) then click on the button Execute.

**GET** /networks/{networkID}/nodes/status Get the status of your blockchain nodes

Returns the operational status of all your nodes on the network

**Parameters**

Name	Description
<b>networkID</b> * required	ID of the IBP network string (path)

networkID - ID of the IBP network

**Execute**

**Responses**

Code	Description
200	<b>OK</b>

Example Value | Model

```
{
  "exampleNodeID": {
    "status": "running",
    "node_type": "fabric-ca"
  }
}
```

Response content type application/json

## IBM Blockchain

The screenshot shows the API documentation for the `/networks/{networkID}/nodes/status` endpoint. It includes a description, parameters, responses, and an execute button.

**GET /networks/{networkID}/nodes/status** Get the status of your blockchain nodes

Returns the operational status of all your nodes on the network

**Parameters**

Name	Description
<b>networkID</b> * required string (path)	ID of the IBP network nde911e8ac6e24386840db89fcb32438f

**Responses**

Code	Description
200	<b>OK</b>  Example Value   Model { "exampleNodeID": f

**Execute**

You will get the list of nodes with their type and their status as shown in the following picture:

The screenshot shows the curl command and its response for the `/networks/{networkID}/nodes/status` endpoint.

**Curl**

```
curl -X GET "https://blockchain-starter.eu-gb.bluemix.net/api/v1/networks/nde911e8ac6e24386840db89fcb32438f/nodes/status" -H "accept: application/json" -H "authorization: Basic b3JnMTpTdFkwS01fbjk1cm9ZemNlbkJFnLRVZTNlQVdBS1NXN3ZYX19UTXL2Q21fdnFfb05PSW14Q29qcWFadGVuVDh4"
```

**Request URL**

```
https://blockchain-starter.eu-gb.bluemix.net/api/v1/networks/nde911e8ac6e24386840db89fcb32438f/nodes/status
```

**Server response**

Code	Details
200	<b>Response body</b> { "org1-ca": { "status": "running", "node_type": "fabric-ca" }, "org1-peer1": { "status": "running", "node_type": "fabric-peer" }, "orderer1": { "status": "running", "node_type": "fabric-orderer" } }  <b>Response headers</b> access-control-allow-origin: * access-control-expose-headers: Location cache-control: no-store connection: Keep-Alive content-security-policy: frame-ancestors *.bluemix.net content-type: application/json; charset=utf-8 date: Fri, 12 Oct 2018 10:00:47 GMT

## 5.3 Get the list of CA users

Now, let's see the users:

**CA Users** ▾

**GET** /networks/{networkID}/users Get the list of CA users

Retreive the list of registered users (enroll IDs) on the CA.

Parameters

Cancel

Name	Description
<b>networkID</b> * required string (path)	ID of the IBP network nde911e8ac6e24386840db89fcb32438f

Execute Clear

Responses

Response content type application/json

Curl

```
curl -X GET "https://blockchain-starter.eu-gb.bluemix.net/api/v1/networks/nde911e8ac6e24386840db89fcb32438f/users" -H "accept: application/json" -H "authorization: Basic b3JnMTpTdFkwS01fbjkIcm9ZemNlbkJFNLRVZTNLQVdBS1NXN3ZYX19UTXL2Q21fdnFb05PSW14Q29qcWFadGVuVdh4"
```



## The result

```
{
  "message": "ok",
  "ca": "org1-ca",
  "msp_id": "org1",
  "client_users": [
    {
      "id": "admin",
      "type": "client",
      "affiliation": "",
      "attrs": [
        {
          "name": "hf.Revoker",
          "value": "1"
        },
        {
          "name": "hf.GenCRL",
          "value": "1"
        },
        {
          "name": "hf.Registrar.Attributes",
          "value": "*"
        },
        {
          "name": "hf.AffiliationMgr",
          "value": "1"
        },
        {
          "name": "hf.Registrar.Roles",
          "value": "client,user,peer"
        },
        {
          "name": "hf.Registrar.DelegateRoles",
          "value": "client,user,peer"
        }
      ],
      "max_enrollments": -1
    }
  ]
}
```

## IBM Blockchain

```
"id": "peer1",
"type": "client",
"affiliation": "org1",
"attrs": [
  {
    "name": "hf.EnrollmentID",
    "value": "peer1",
    "ecert": true
  },
  {
    "name": "hf.Type",
    "value": "client",
    "ecert": true
  },
  {
    "name": "hf.Affiliation",
    "value": "org1",
    "ecert": true
  }
],
"max_enrollments": 1
},
"ibp_internal_users": [
{
  "id": "ibmuser",
  "type": "client",
  "affiliation": "",
  "attrs": [
    {
      "name": "hf.Registrar.DelegateRoles",
      "value": "client,user,peer,orderer"
    },
    {
      "name": "hf.Revoker",
      "value": "1"
    },
    {
      "name": "hf.IntermediateCA",
      "value": "1"
    },
    {
      "name": "hf.GenCRL",
      "value": "1"
    },
    {
      "name": "hf.Registrar.Attributes",
      "value": "*"
    },
    {
      "name": "hf.AffiliationMgr",
      "value": "1"
    },
    {
      "name": "hf.Registrar.Roles",
      "value": "client,user,peer,orderer"
    }
  ],
  "max_enrollments": -1
},
{
  "id": "org1_admin",
  "type": "client",
  "affiliation": "org1",
  "attrs": [
    {
      "name": "hf.EnrollmentID",
      "value": "org1_admin",
    }
  ]
}],
```

```
        "eceipt": true
    },
    {
        "name": "hf.Type",
        "value": "client",
        "eceipt": true
    },
    {
        "name": "hf.Affiliation",
        "value": "org1",
        "eceipt": true
    }
],
"max_enrollments": 1
}
]
```

## 6 Smartcontract execution

We are going to execute the smartcontract in the IBM Blockchain Platform environment. We will use a NodeJS application to invoke the transactions.

### 6.1 Preparation

First, we need to retrieve the connection profile of the IBM Blockchain Platform.

Go to the Blockchain monitor Overview view, and click on the Connection Profile button.

TYPE	NAME	STATUS	ACTIONS
Orderer	orderer	Running	⋮
CA	org1-ca	Running	↻ ⏪ ⋮
Peer	org1-peer1	Running	↻ ⏪ ⋮

Then click on download. Save the file as network-profile.json.

Open it with an editor. Go to the certificateAuthorities section, then look at

- the org1-ca url
- the enrollId and enrollSecret
- the caName

as shown in the following extract:

```
"certificateAuthorities": {
    "org1-ca": {
        "url": "https://nde911e8ac6e24386840db89fcb32438f-org1-
ca.uk02.blockchain.ibm.com:31011",
        "httpOptions": {
            "verify": true
        },
        "tlsCACerts": {
            "pem": "-----BEGIN CERTIFICATE...-----END CERTIFICATE-----\r\n"
        }
    }
}
```

```

},
"registrar": [
{
    "enrollId": "admin",
    "enrollSecret": "xxxxxxxxxx",
    "x-affiliations": [
        "org1",
        "org1.department1",
        "org1.department2",
        "org2",
        "org2.department1"
    ]
},
],
"caName": "org1CA",
"x-mspid": "org1"

```

This information will be used to enroll the admin user of the application.

## 6.2 Run the smartcontract

We will use a simple NodeJS client application to run the transactions.

Load the client application

```
cd $GOPATH/src/labhf
git clone http://github.com/ovallod/labhf-cli
```

Copy the network-profile.json in the config folder:

```
cd $GOPATH/src/labhf/labhf-cli
cp <path-to-network-profile.json>/network-profile.json config/
```

Install the node packages required for this client:

```
cd $GOPATH/src/labhf/labhf-cli
npm install
```

Enroll the admin of the application and retrieve its certificates which will be stored in the hfc-key-store folder:

```
node enrollAdminNetwork.js
```

The result looks like:

```
Found organization: org1 and ca name: org1-ca
Enrolling using enrollmentId: admin and EnrollmentSecret: xxxxxxxx
(node:22870) DeprecationWarning: grpc.load: Use the @grpc/proto-loader module with grpc.loadPackageDefinition instead
Successfully enrolled admin user "admin" with msp: "org1"
Assigned the admin user to the fabric client
:{"name":"admin","mspid":"org1","roles":null,"affiliation":"","enrollmentSecret":"","enrollment":{"signingIdentity":"...","identity":{"certificate":"-----BEGIN CERTIFICATE-----\n-----END CERTIFICATE-----\n"}}
```

Now we are going to run the transaction OrderShipment with the user admin.

First, update the file parameters.json in the config folder with the user, chaincode name, the transaction name, and the arguments of the transaction:

```
{
```

```

"user" : "admin",
"chaincodeId" : "labhlf-<xx>-sc",
"transactionName": "OrderShipment",
"transactionArgs" : "{\"packageID\":\"P1\",\"description\":\"Package for product P001\",\"destination\": \"Montpellier, FRANCE, 34006\"}"
}

```

```
node invokeNetwork.js
```

The expected result is:

```
(node:24232) DeprecationWarning: grpc.load: Use the @grpc/proto-loader module with grpc.loadPackageDefinition instead
Successfully loaded admin from persistence
Assigning transaction_id: 36994a8ae570b9aabf5bd465d6572ec8554da9bfc9883c8b012e17ea3d4ab046
Transaction proposal was good
Successfully sent Proposal and received ProposalResponse: Status - 200, message - ""
Getting event hub
The transaction has been committed on peer nde911e8ac6e24386840db89fcb32438f-org1-peer1.uk02.blockchain.ibm.com:31002
Send transaction promise and event listener promise have completed
Successfully sent transaction to the orderer.
Successfully committed the change to the ledger by the peer
```

## 6.3 Check the transaction in the StarterPlan

Go back to the Blockchain monitor, open the channel view and click on the channel line to access the statistic view. We observe the counter of blocks has increased. And you can see the detail of the transactions, clicking on the block line, then clicking on the actions button then selecting “View detail”.

The screenshot shows the IBM Blockchain Monitor interface. On the left, there's a sidebar with navigation links: 'Everest Network - VZ6', 'MY NETWORK', 'Overview', 'Members', 'Channels' (which is selected and highlighted in blue), 'Notifications', 'Certificate Authority', 'APIs', 'MY CODE', 'Develop code', and 'Install code'. The main content area is titled 'defaultchannel' with the subtitle 'View channel activity and configuration'. It features a 'Channel Overview' section with four key metrics: 'TOTAL BLOCKS' (10), 'TIME SINCE LAST TRANSACTION' (9 hrs), 'RECENT INSTANTIATIONS' (1), and 'RECENT INVOCATIONS' (8). Below this is a table showing transaction details:

TIME	BLOCK NUMBER	INSTANTIATIONS	INVOCATIONS
10/23/18 23:03 GMT+0200	9	0	1

At the bottom, there's another table for chaincode invocations:

DATE	TYPE	UUID	CHAINCODE ID(S)	ACTIONS
10/23/18 23:03 GMT+0200	INVOKE	36cbc487f5b387d1e4add876043a3106df28753d44ac0bc7c6ab4b4882f9b642	labhlf-ov-sc	⋮

---

## Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries.

Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service. IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation

Licensing

2-31 Roppongi 3-chome, Minato-ku

Tokyo 106-0032, Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES**

CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM

products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental. All references to fictitious companies or individuals are used for illustration purposes only.

**COPYRIGHT LICENSE:**

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

---

## Appendix A. Trademarks and copyrights

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	AIX	CICS	ClearCase	ClearQuest	Cloudscape
Cube Views	DB2	developerWorks	DRDA	IMS	IMS/ESA
Informix	Lotus	Lotus Workflow	MQSeries	OmniFind	
Rational	Redbooks	Red Brick	RequisitePro	System i	
<i>System z</i>	<i>Tivoli</i>	<i>WebSphere</i>	<i>Workplace</i>	<i>System p</i>	

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of The Minister for the Cabinet Office, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

## **NOTES**

## **NOTES**



---

© Copyright IBM Corporation 2016.

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.

IBM, the IBM logo and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).



Please Recycle

---