IBM Blockchain
Proof of Technology
Blockchain Explored

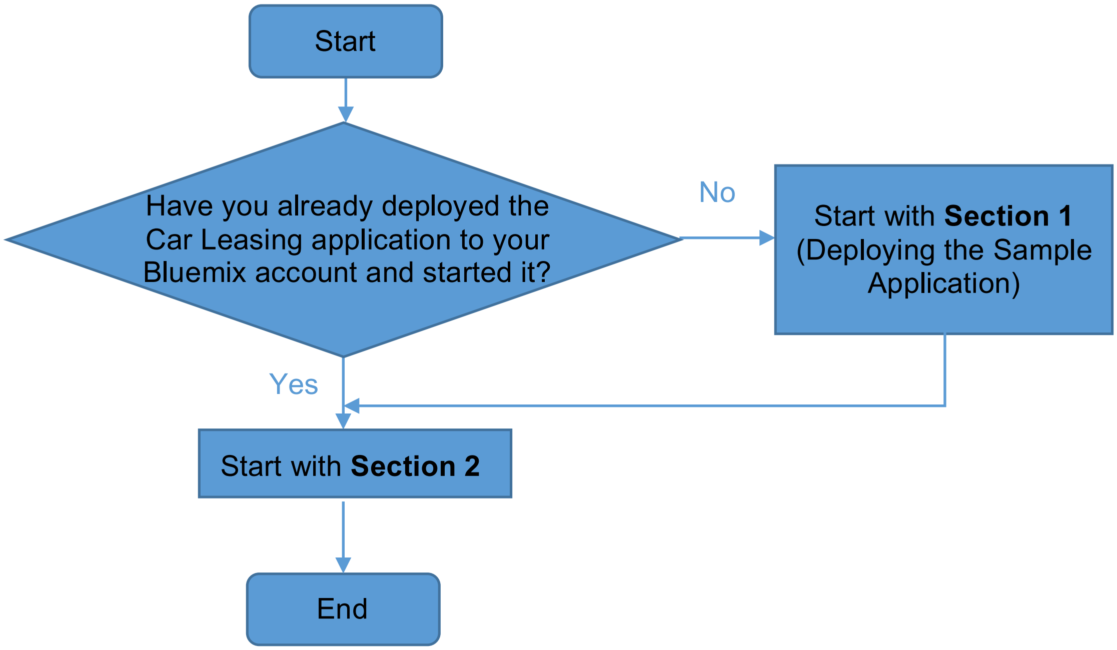
Lab Two – Bluemix - Exercises

Contents

Overview Introduction to the Lab

The purpose of this lab is to introduce you to the IBM Blockchain service on Bluemix. We will build on the car leasing scenario that was introduced in the "Blockchain Explained" lab.

If you are using your own Bluemix account and have already completed the previous lab, you will have already deployed the car leasing application to your account. You can skip section 1 and re-use your existing application:


Start
Have you already deployed the Car Leasing application to your Bluemix account and started it?
No
Start with **Section 1** (Deploying the Sample Application)
Yes
Start with **Section 2**
End

Section 1. Deploying the sample application

In this section we will use Bluemix to deploy a copy of the car leasing demo application.

1.1. Create a Blockchain Service

__1. Open a web browser and go to www.bluemix.net.

> It is recommended to use Firefox or Chrome.
>
> Problems can generally be resolved by clearing the browser's cache and cookies, or running the browser in private mode.

__2. Select CATALOG from the top bar

__3. Scroll down to the network section and click **Blockchain**

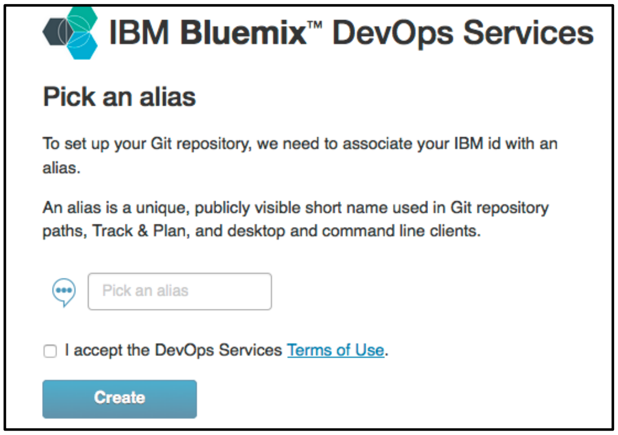
__4. Review the service description and information about the service.

__5. Click VIEW DOCS and learn about the process of creating a blockchain environment.

__6. Click 'Sample Apps and Tutorials' on the right of the page to view the available apps.

__7. Click Deploy to Bluemix against the car lease demo. Log in to Bluemix again if necessary.

> The first time a Bluemix ID creates a sample, a new DevOps alias is required. **Pick a unique ID** and click acceptance of the terms, then click create. For Proof-of-Technology workshops, use the first part of the email address, for example ibmpot000101 (without the '+' character). Then click continue on the following page.



IBM Bluemix™ DevOps Services

Pick an alias

To set up your Git repository, we need to associate your IBM id with an alias.

An alias is a unique, publicly visible short name used in Git repository paths, Track & Plan, and desktop and command line clients.

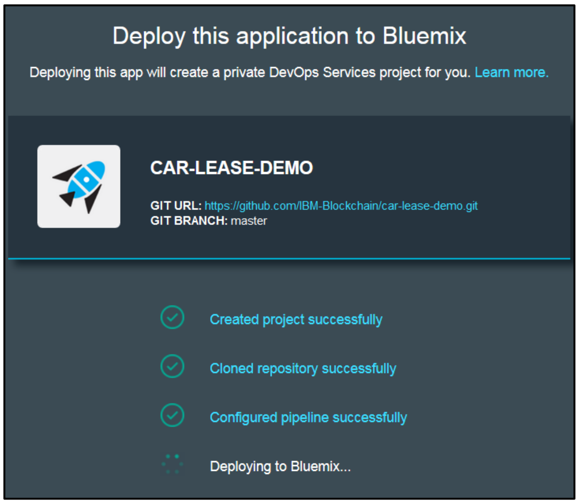
Pick an alias

□ I accept the DevOps Services Terms of Use.

Create

__8. Leave the App Name, Region, Organization and Space default and click DEPLOY. (You might first need to wait a few seconds for the default field values to be populated.)

Clicking Deploy will cause the car leasing demo to be deployed into your Bluemix environment, and may take a couple of minutes to complete.



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

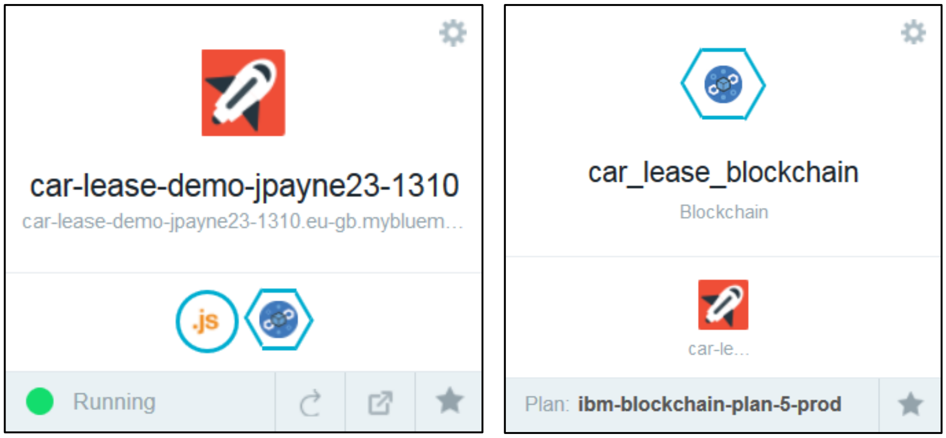
✓ Created project successfully

✓ Cloned repository successfully

✓ Configured pipeline successfully

Deploying to Bluemix...

__9. Once you see the 'Success!' message click DASHBOARD to see the new car leasing application (and associated Blockchain service) you created.



car-lease-demo-jpayne23-1310

car-lease-demo-jpayne23-1310.eu-gb.mybluem...

● Running

car_lease_blockchain

Blockchain

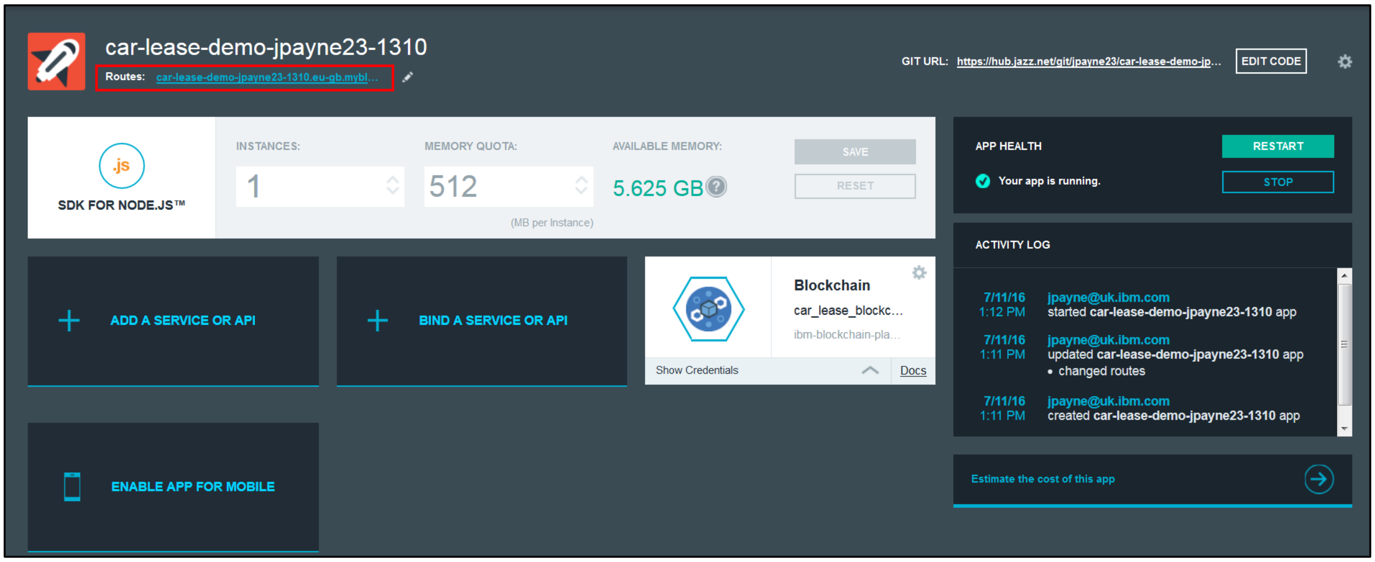
car-le...

Plan: ibm-blockchain-plan-5-prod

__10. Click the Node.js application's icon in the dashboard (**your icon may vary**)

This will show you information about the application, including the memory that it is consuming and activity log.

__11. Click the 'Routes' URL (something like **Routes:** car-lease-demo-jpayne23-1310.eu-gb.mybl…) in order to run the scenario. This will load a webpage which is served from the application.



car-lease-demo-jpayne23-1310

Routes: car-lease-demo-jpayne23-1310.eu-gb.mybl…

GIT URL: https://hub.jazz.net/git/jpayne23/car-lease-demo-jp…

INSTANCES: 1

MEMORY QUOTA: 512 (MB per Instance)

AVAILABLE MEMORY: 5.625 GB

Your app is running.

Blockchain
car_lease_blockc…
ibm-blockchain-pla…

ACTIVITY LOG

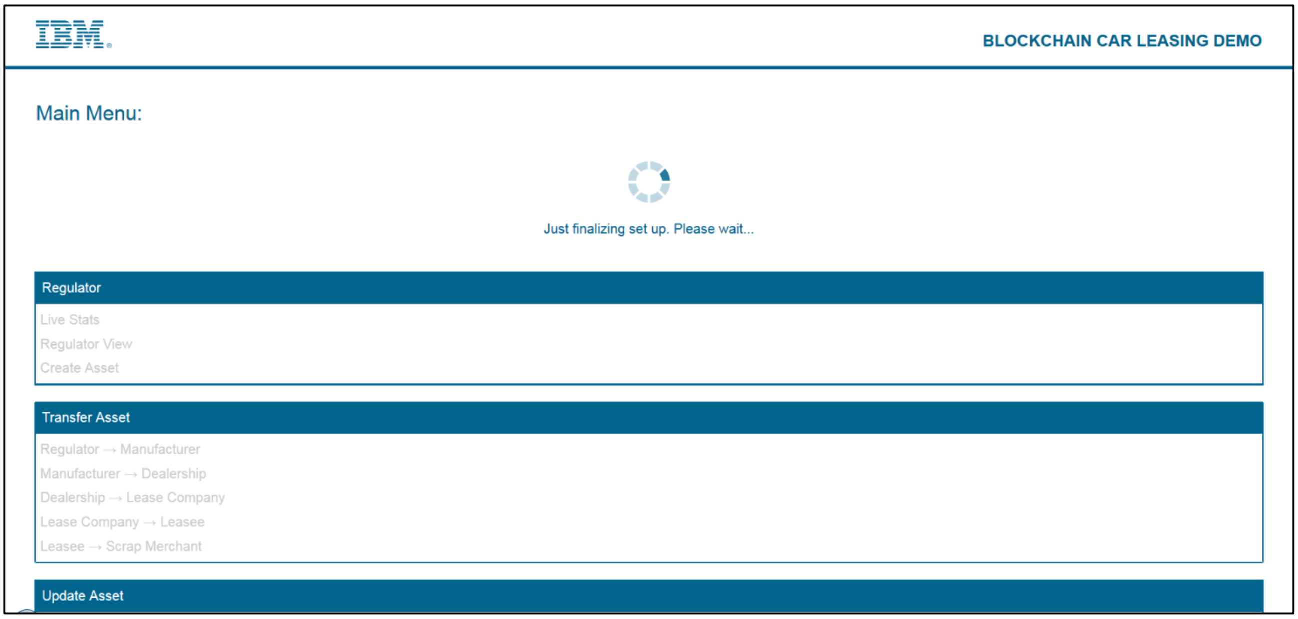
7/11/16 1:12 PM jpayne@uk.ibm.com started **car-lease-demo-jpayne23-1310** app

7/11/16 1:11 PM jpayne@uk.ibm.com updated **car-lease-demo-jpayne23-1310** app
- changed routes

7/11/16 1:11 PM jpayne@uk.ibm.com created **car-lease-demo-jpayne23-1310** app

Estimate the cost of this app

__12. You may see the page below, this means the application isn't ready to be used yet. Just wait until you see the screen shown in the next step.



Main Menu:

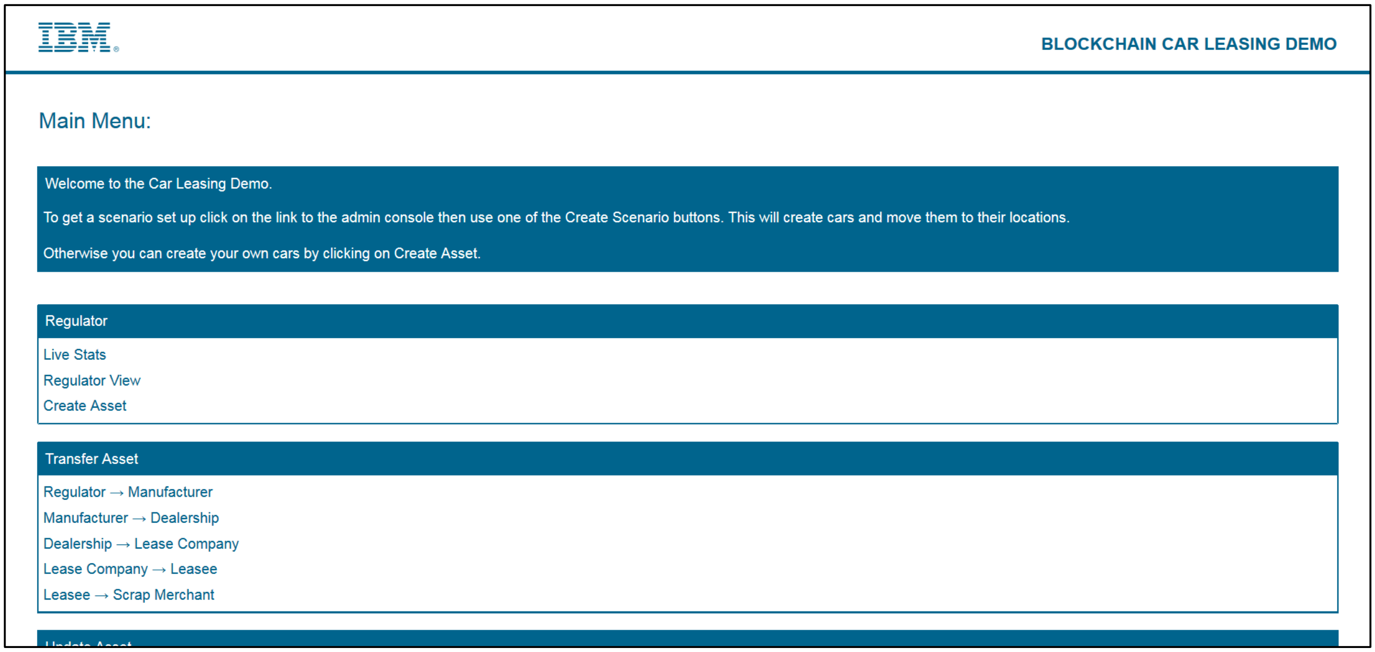
Just finalizing set up. Please wait...

Regulator
- Live Stats
- Regulator View
- Create Asset

Transfer Asset
- Regulator → Manufacturer
- Manufacturer → Dealership
- Dealership → Lease Company
- Lease Company → Leasee
- Leasee → Scrap Merchant

Update Asset

__13. A message will appear once the application is ready to be used and the home page will look the same as below.

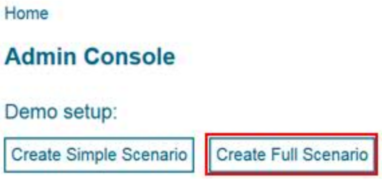


Welcome to the Car Leasing Demo.

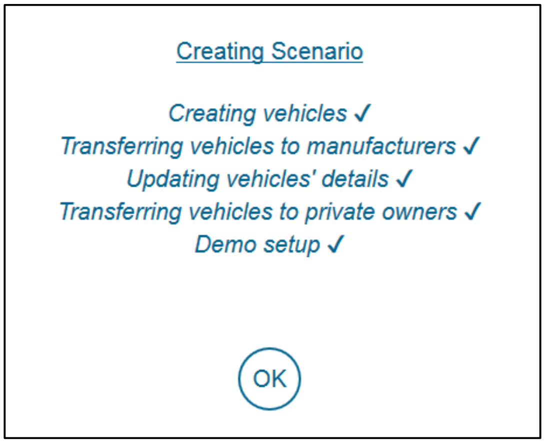
To get a scenario set up click on the link to the admin console then use one of the Create Scenario buttons. This will create cars and move them to their locations.

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

__14. Click ‘Admin Console’ and then ‘Create Simple Scenario’ to load the initial set of assets into the Blockchain. This may take several minutes to complete.



__15. The scenario setup is complete when ‘Demo setup’ is displayed.



Creating vehicles ✓
Transferring vehicles to manufacturers ✓
Updating vehicles' details ✓
Transferring vehicles to private owners ✓
Demo setup ✓

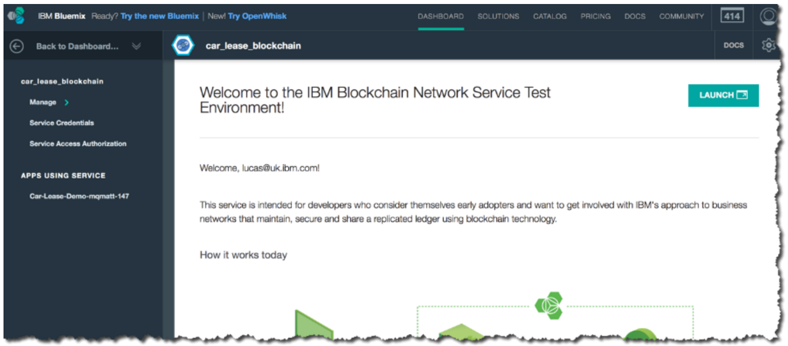
Section 2. Managing the sample application

In this section we will use the tools available inside the Bluemix environment to view and manage the Blockchain.

2.1. View the components of the Blockchain service

__1. In Bluemix, select DASHBOARD to view the car leasing application.

__2. Click on the service icon for your new blockchain service in the dashboard. This will take you to the service welcome screen.



Welcome to the IBM Blockchain Network Service Test Environment!

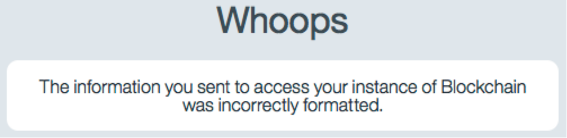
Welcome, lucas@uk.ibm.com!

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.

How it works today

__3. Review the details and select LAUNCH to launch the service console.

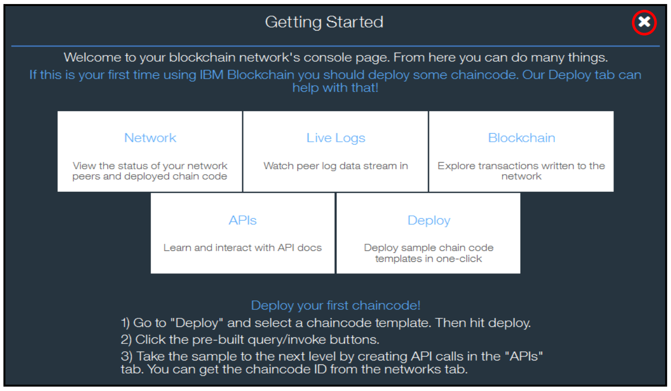
If you see the following error, then try changing to a Firefox or Chrome browser:



Whoops

The information you sent to access your instance of Blockchain was incorrectly formatted.

__4. Close the pop-up showing information about the sections. We'll look at these in more detail throughout this lab.



Getting Started

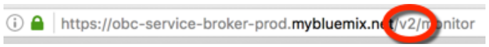
Welcome to your blockchain network's console page. From here you can do many things.
If this is your first time using IBM Blockchain you should deploy some chaincode. Our Deploy tab can help with that!

Network
View the status of your network peers and deployed chain code
Live Logs
Watch peer log data stream in
Blockchain
Explore transactions written to the network
APIs
Learn and interact with API docs
Deploy
Deploy sample chain code templates in one-click

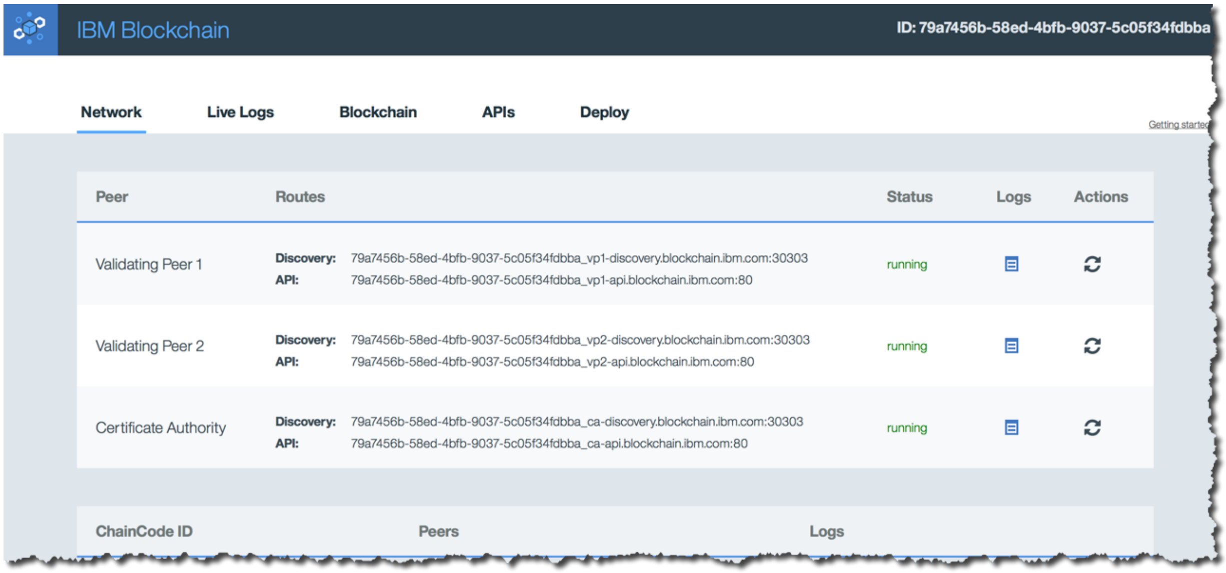
Deploy your first chaincode!
1) Go to "Deploy" and select a chaincode template. Then hit deploy.
2) Click the pre-built query/invoke buttons.
3) Take the sample to the next level by creating API calls in the "APIs" tab. You can get the chaincode ID from the networks tab.

__5. This takes you to the monitor page, with the Network tab selected.

If you see fewer than five tabs on this page, try changing the version in the URL to ‘v2’ and reloading the page. For example:



https://obc-service-broker-prod.mybluemix.net/v2/monitor



IBM Blockchain

ID: 79a7456b-58ed-4bfb-9037-5c05f34fdbba

Peer	Routes	Status	Logs	Actions
Validating Peer 1	**Discovery:** 79a7456b-58ed-4bfb-9037-5c05f34fdbba_vp1-discovery.blockchain.ibm.com:30303 **API:** 79a7456b-58ed-4bfb-9037-5c05f34fdbba_vp1-api.blockchain.ibm.com:80	running	▤	↻
Validating Peer 2	**Discovery:** 79a7456b-58ed-4bfb-9037-5c05f34fdbba_vp2-discovery.blockchain.ibm.com:30303 **API:** 79a7456b-58ed-4bfb-9037-5c05f34fdbba_vp2-api.blockchain.ibm.com:80	running	▤	↻
Certificate Authority	**Discovery:** 79a7456b-58ed-4bfb-9037-5c05f34fdbba_ca-discovery.blockchain.ibm.com:30303 **API:** 79a7456b-58ed-4bfb-9037-5c05f34fdbba_ca-api.blockchain.ibm.com:80	running	▤	↻

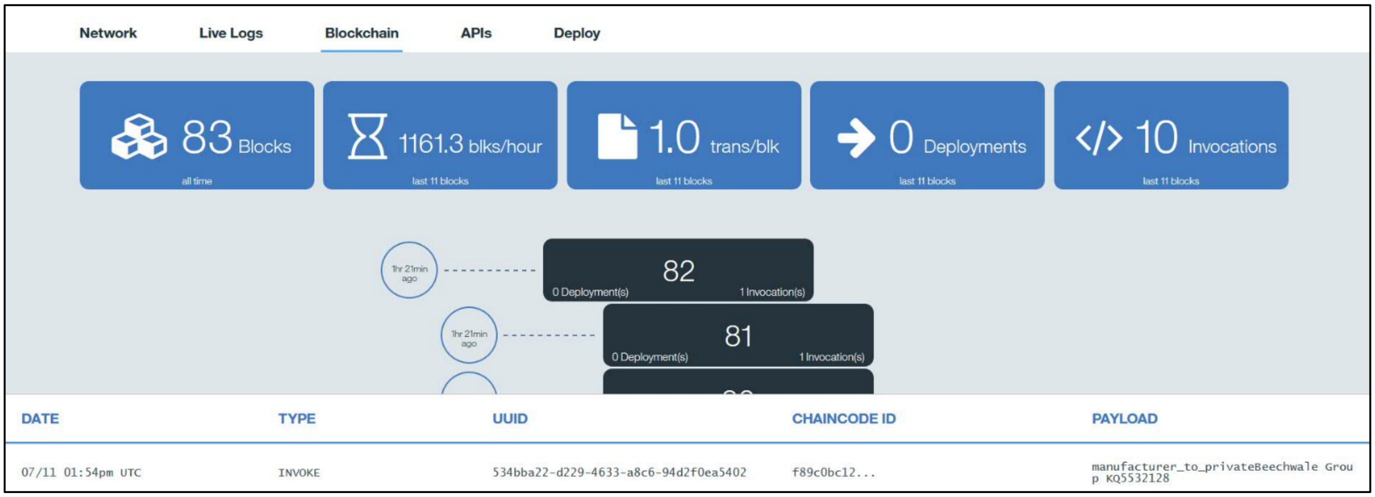
ChainCode ID	Peers	Logs

This view confirms that two validating peers and a certificate authority are running under the service you created.

2.2. View the Blockchain Explorer

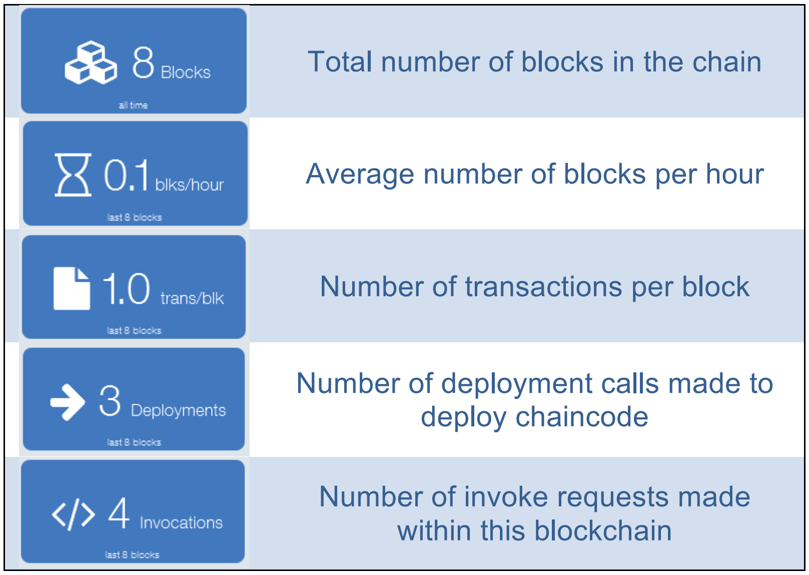
The Blockchain explorer is a visual representation of the state of the Blockchain.

__1. Click the ‘Blockchain’ tab at the top of the page.



DATE	TYPE	UUID	CHAINCODE ID	PAYLOAD
07/11 01:54pm UTC	INVOKE	534bba22-d229-4633-a8c6-94d2f0ea5402	f89c0bc12...	manufacturer_to_privateBeechwale Group KQ5532128

The icons show:

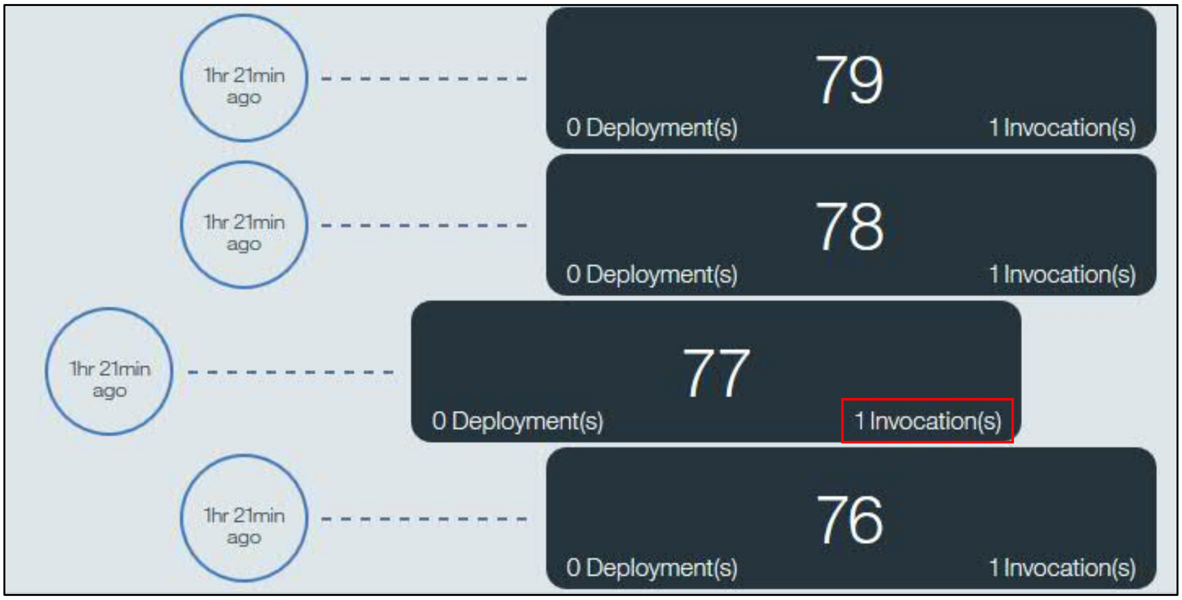


8 Blocks all time	Total number of blocks in the chain
0.1 blks/hour last 8 blocks	Average number of blocks per hour
1.0 trans/blk last 8 blocks	Number of transactions per block
3 Deployments last 8 blocks	Number of deployment calls made to deploy chaincode
4 Invocations last 8 blocks	Number of invoke requests made within this blockchain

Each block contains a set of transactions. In Hyperledger, a transaction is the record of the request to interact with chaincode (a smart contract). Two important transaction types are:

- *INVOKE*: The request to invoke a piece of chaincode (for example, invoke the chaincode to transfer the ownership of a car)
- *DEPLOY*: The request to deploy a piece of chaincode across all validating peers, so that it can be executed at a later date.

Other request types exist (e.g. query, update, terminate). Not all request types are recorded on the Blockchain.



1hr 21min ago
79
0 Deployment(s) 1 Invocation(s)

1hr 21min ago
78
0 Deployment(s) 1 Invocation(s)

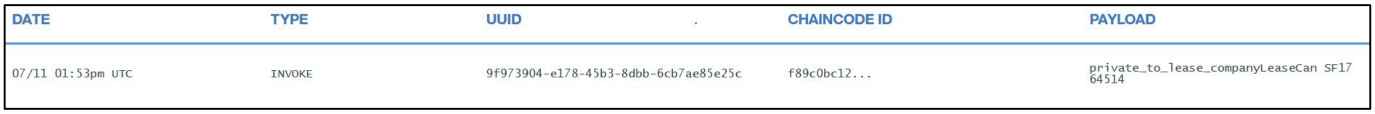
1hr 21min ago
77
0 Deployment(s) 1 Invocation(s)

1hr 21min ago
76
0 Deployment(s) 1 Invocation(s)

The blocks also include when that block was committed to the Blockchain.

__2. Click on a block that contains at least one invocation request.

__3. Look through the list of transactions that are contained within the block.



DATE	TYPE	UUID	CHAINCODE ID	PAYLOAD
07/11 01:53pm UTC	INVOKE	9f973904-e178-45b3-8dbb-6cb7ae85e25c	f89c0bc12...	private_to_lease_companyLeaseCan SF1764514

Each line of information is a transaction stored within the block. A block may contain multiple transactions but in this demo there will often only be one transaction per block due to the low frequency of transactions being made. The information displayed is:

Date	The date the transaction was submitted.
Type	The type of transaction taking place (e.g. INVOKE or DEPLOY).
UUID	The unique identifier for each transaction.
Chaincode ID	Refers to the chaincode that is being invoked or deployed.
Payload	The input parameters to the chaincode.

__4. Repeat this for other blocks to understand how the transactions are stored.

ℹ	When the Blockchain is initialised for the car leasing application, the first two blocks in the chain usually contain ‘Deploy’ transactions, where the chaincode is deployed to the validating peers. View these blocks If you’re willing to scroll down the Blockchain explorer that far!

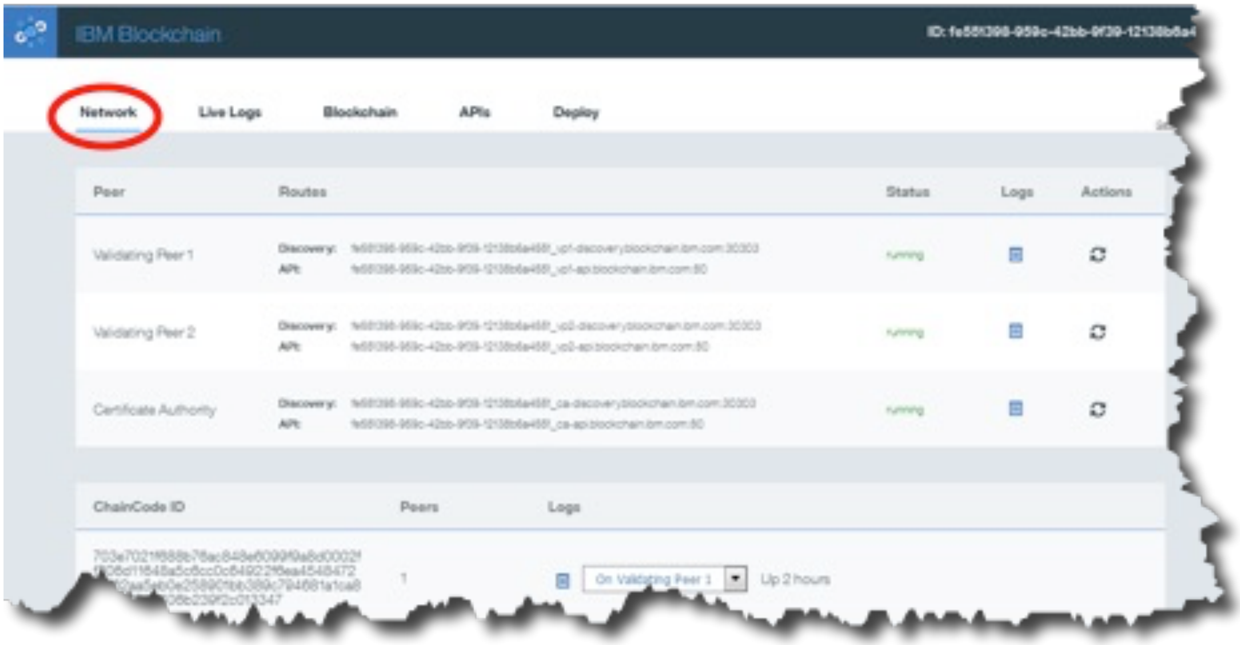
2.3. Understanding the Blockchain Peers

We are now going to review the logs associated with the peers. This is useful for understanding how the Blockchain works, and for diagnosing problems.

There are two ways of accessing the logs of the peers:

- A “Logs” button on the Network tab. This is useful for downloading log files from the peers for offline analysis
- The “Live Logs” tab that shows you what the peers are doing now.

__1. Click on the ‘Network’ tab at the service page.



Peer	Routes	Status	Logs	Actions
Validating Peer 1	Discovery: fe581398-959c-42bb-9f39-12138b6a4681_vp1-discovery.blockchain.ibm.com:30303 API: fe581398-959c-42bb-9f39-12138b6a4681_vp1-api.blockchain.ibm.com:80	running	▤	⟳
Validating Peer 2	Discovery: fe581398-959c-42bb-9f39-12138b6a4681_vp2-discovery.blockchain.ibm.com:30303 API: fe581398-959c-42bb-9f39-12138b6a4681_vp2-api.blockchain.ibm.com:80	running	▤	⟳
Certificate Authority	Discovery: fe581398-959c-42bb-9f39-12138b6a4681_ca-discovery.blockchain.ibm.com:30303 API: fe581398-959c-42bb-9f39-12138b6a4681_ca-api.blockchain.ibm.com:80	running	▤	⟳

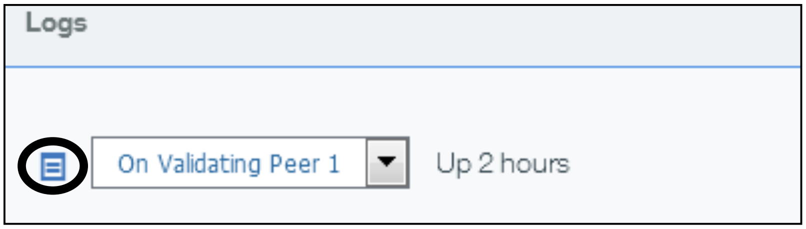
ChainCode ID	Peers	Logs
703e7021f888b76ac848e6099f9a8d0002f f806d11848a5c6cc0c64922f6ea4548472 2aa5eb0e25890fbb389c794681a1ca8 06b239f2c013347	1	▤ On Validating Peer 1 ▾ Up 2 hours

Here we can see that this Blockchain contains two validating peers and a Certificate Authority. The table underneath shows that there are two chaincode applications deployed to this network.

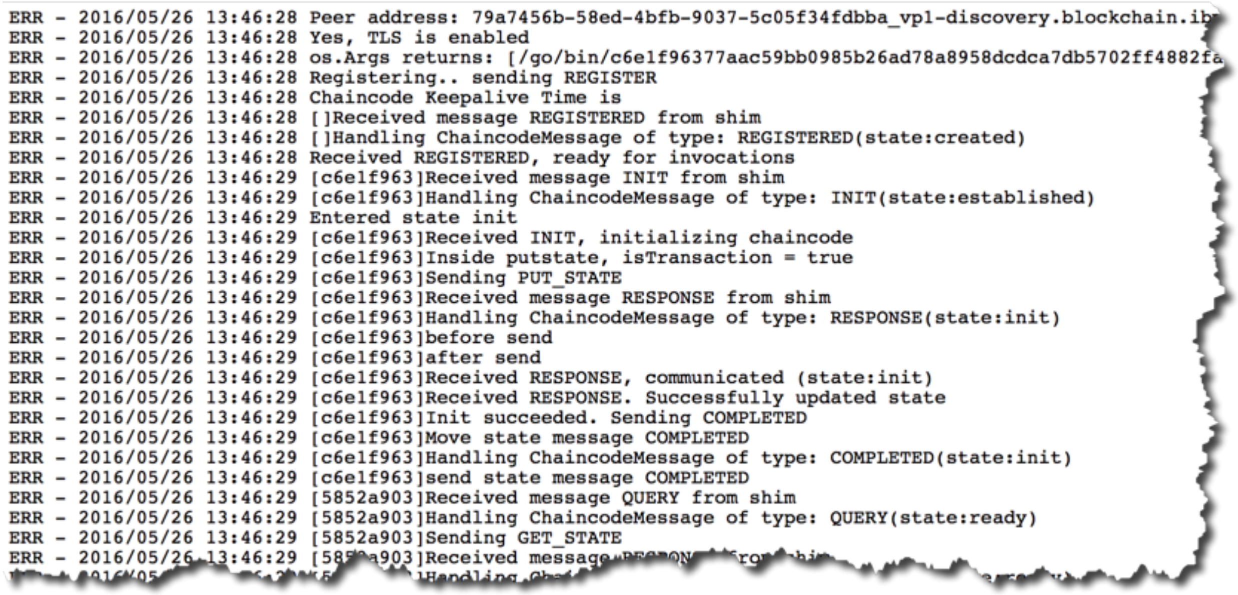
Requests to invoke chaincode (including the method name and any input parameters) are replicated onto every validating node, and when a block is created every validating node will execute the chaincode independently. (The validating peers then attempt to achieve consensus over any changes proposed to the world state as a result of running this chaincode, and as a consequence will persist or discard the changes.)

By looking at the logs for each peer you can verify that every node has executed every transaction.

__2. Select the peer for which you wish to see the logs and click on the ☰ icon.

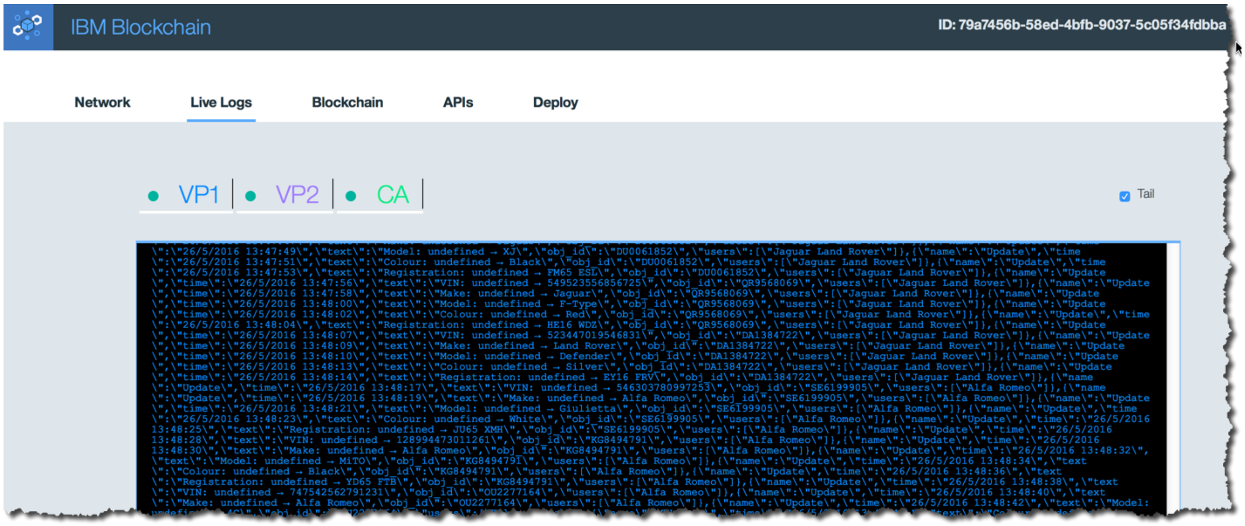


__3. This will show the logs for the selected peer in a new tab.



```
ERR - 2016/05/26 13:46:28 Peer address: 79a7456b-58ed-4bfb-9037-5c05f34fdbba_vp1-discovery.blockchain.ib
ERR - 2016/05/26 13:46:28 Yes, TLS is enabled
ERR - 2016/05/26 13:46:28 os.Args returns: [/go/bin/c6e1f96377aac59bb0985b26ad78a8958dcdca7db5702ff4882fa
ERR - 2016/05/26 13:46:28 Registering.. sending REGISTER
ERR - 2016/05/26 13:46:28 Chaincode Keepalive Time is
ERR - 2016/05/26 13:46:28 []Received message REGISTERED from shim
ERR - 2016/05/26 13:46:28 []Handling ChaincodeMessage of type: REGISTERED(state:created)
ERR - 2016/05/26 13:46:28 Received REGISTERED, ready for invocations
ERR - 2016/05/26 13:46:29 [c6e1f963]Received message INIT from shim
ERR - 2016/05/26 13:46:29 [c6e1f963]Handling ChaincodeMessage of type: INIT(state:established)
ERR - 2016/05/26 13:46:29 Entered state init
ERR - 2016/05/26 13:46:29 [c6e1f963]Received INIT, initializing chaincode
ERR - 2016/05/26 13:46:29 [c6e1f963]Inside putstate, isTransaction = true
ERR - 2016/05/26 13:46:29 [c6e1f963]Sending PUT_STATE
ERR - 2016/05/26 13:46:29 [c6e1f963]Received message RESPONSE from shim
ERR - 2016/05/26 13:46:29 [c6e1f963]Handling ChaincodeMessage of type: RESPONSE(state:init)
ERR - 2016/05/26 13:46:29 [c6e1f963]before send
ERR - 2016/05/26 13:46:29 [c6e1f963]after send
ERR - 2016/05/26 13:46:29 [c6e1f963]Received RESPONSE, communicated (state:init)
ERR - 2016/05/26 13:46:29 [c6e1f963]Received RESPONSE. Successfully updated state
ERR - 2016/05/26 13:46:29 [c6e1f963]Init succeeded. Sending COMPLETED
ERR - 2016/05/26 13:46:29 [c6e1f963]Move state message COMPLETED
ERR - 2016/05/26 13:46:29 [c6e1f963]Handling ChaincodeMessage of type: COMPLETED(state:init)
ERR - 2016/05/26 13:46:29 [c6e1f963]send state message COMPLETED
ERR - 2016/05/26 13:46:29 [5852a903]Received message QUERY from shim
ERR - 2016/05/26 13:46:29 [5852a903]Handling ChaincodeMessage of type: QUERY(state:ready)
ERR - 2016/05/26 13:46:29 [5852a903]Sending GET_STATE
ERR - 2016/05/26 13:46:29 [5852a903]Received message RESPONSE from shim
```

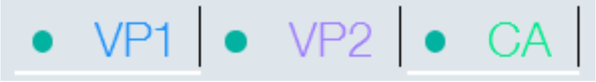
__4.	Click on the ‘Live Logs’ tab on the service page.



```
\":\"26/5/2016 13:47:49\",\"text\":\"Model: undefined → XJ\",\"obj_id\":\"DU0061852\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update\",\"time
\":\"26/5/2016 13:47:51\",\"text\":\"Colour: undefined → Black\",\"obj_id\":\"DU0061852\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update\",\"time
\":\"26/5/2016 13:47:53\",\"text\":\"Registration: undefined → FM65 ESL\",\"obj_id\":\"DU0061852\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:47:56\",\"text\":\"VIN: undefined → 549523556856725\",\"obj_id\":\"QR9568069\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:47:58\",\"text\":\"Make: undefined → Jaguar\",\"obj_id\":\"QR9568069\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:48:00\",\"text\":\"Model: undefined → F-Type\",\"obj_id\":\"QR9568069\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:48:02\",\"text\":\"Colour: undefined → Red\",\"obj_id\":\"QR9568069\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update\",\"time
\":\"26/5/2016 13:48:04\",\"text\":\"Registration: undefined → HE16 WDZ\",\"obj_id\":\"QR9568069\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:48:07\",\"text\":\"VIN: undefined → 523447019546831\",\"obj_id\":\"DA1384722\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:48:09\",\"text\":\"Make: undefined → Land Rover\",\"obj_id\":\"DA1384722\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:48:10\",\"text\":\"Model: undefined → Defender\",\"obj_id\":\"DA1384722\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:48:13\",\"text\":\"Colour: undefined → Silver\",\"obj_id\":\"DA1384722\",\"users\":[\"Jaguar Land Rover\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:48:14\",\"text\":\"Registration: undefined → EY16 FRV\",\"obj_id\":\"DA1384722\",\"users\":[\"Jaguar Land Rover\"]},{\"name
\":\"Update\",\"time\":\"26/5/2016 13:48:17\",\"text\":\"VIN: undefined → 546303780997253\",\"obj_id\":\"SE6199905\",\"users\":[\"Alfa Romeo\"]},{\"name
\":\"Update\",\"time\":\"26/5/2016 13:48:19\",\"text\":\"Make: undefined → Alfa Romeo\",\"obj_id\":\"SE6199905\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update
\",\"time\":\"26/5/2016 13:48:21\",\"text\":\"Model: undefined → Giulietta\",\"obj_id\":\"SE6199905\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time
\":\"26/5/2016 13:48:23\",\"text\":\"Colour: undefined → White\",\"obj_id\":\"SE6199905\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time\":\"26/5/2016
13:48:25\",\"text\":\"Registration: undefined → JU65 XMH\",\"obj_id\":\"SE6199905\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time\":\"26/5/2016
13:48:28\",\"text\":\"VIN: undefined → 128994473011261\",\"obj_id\":\"KG8494791\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time\":\"26/5/2016
13:48:30\",\"text\":\"Make: undefined → Alfa Romeo\",\"obj_id\":\"KG8494791\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time\":\"26/5/2016 13:48:32\",
\"text\":\"Model: undefined → MiTO\",\"obj_id\":\"KG8494791\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time\":\"26/5/2016 13:48:34\",\"text
\":\"Colour: undefined → Black\",\"obj_id\":\"KG8494791\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time\":\"26/5/2016 13:48:36\",\"text
\":\"Registration: undefined → YD65 FTB\",\"obj_id\":\"KG8494791\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time\":\"26/5/2016 13:48:38\",\"text
\":\"VIN: undefined → 747542562791231\",\"obj_id\":\"OU2277164\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time\":\"26/5/2016 13:48:40\",\"text
\":\"Make: undefined → Alfa Romeo\",\"obj_id\":\"OU2277164\",\"users\":[\"Alfa Romeo\"]},{\"name\":\"Update\",\"time\":\"26/5/2016 13:48:42\",\"text\":\"Model:
```

This page shows the same logs that were shown from the Network tab, however these are live updating (if you have ‘Tail’ selected), and you can also see a combined view of multiple validating peers and the Certificate Authority.

__5.	Click on the ‘VP1’, ‘VP2’ and ‘CA’ buttons to toggle on/off the viewing of live logs for each peer.

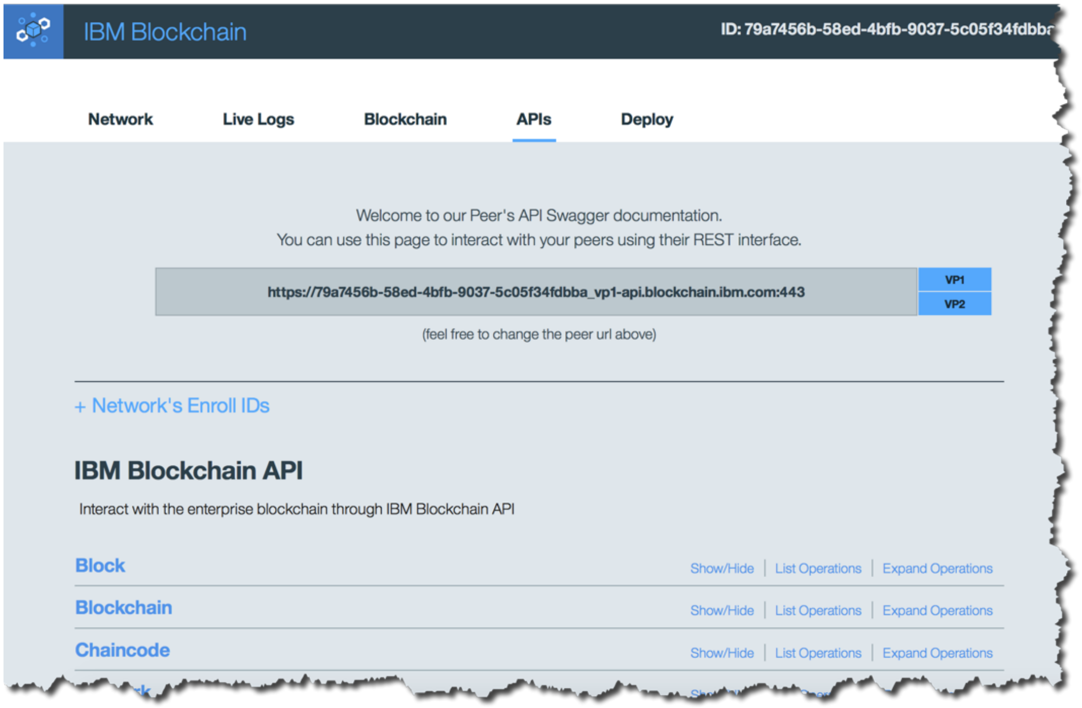


2.4. Interacting with the peers

It is possible to invoke the management APIs that interact directly with the peers. In this section we will be trying out these APIs directly from the Bluemix environment.

Note that the APIs concern *operationally managing* the Blockchain – this is not the same as adding and invoking transactions through chaincode!

__1. Click on the 'APIs' tab on the service page.



IBM Blockchain

ID: 79a7456b-58ed-4bfb-9037-5c05f34fdbba

Network Live Logs Blockchain APIs Deploy

Welcome to our Peer's API Swagger documentation.
You can use this page to interact with your peers using their REST interface.

https://79a7456b-58ed-4bfb-9037-5c05f34fdbba_vp1-api.blockchain.ibm.com:443

VP1
VP2

(feel free to change the peer url above)

+ Network's Enroll IDs

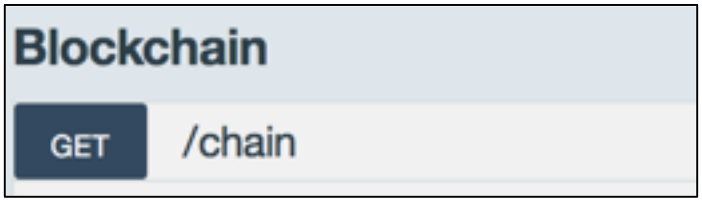
IBM Blockchain API

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

This page allows you to invoke APIs that will directly interrogate and manage the blockchain. First we will use the API interface to query the height of the Blockchain (the number of blocks).

__2. Click the 'Blockchain' section.



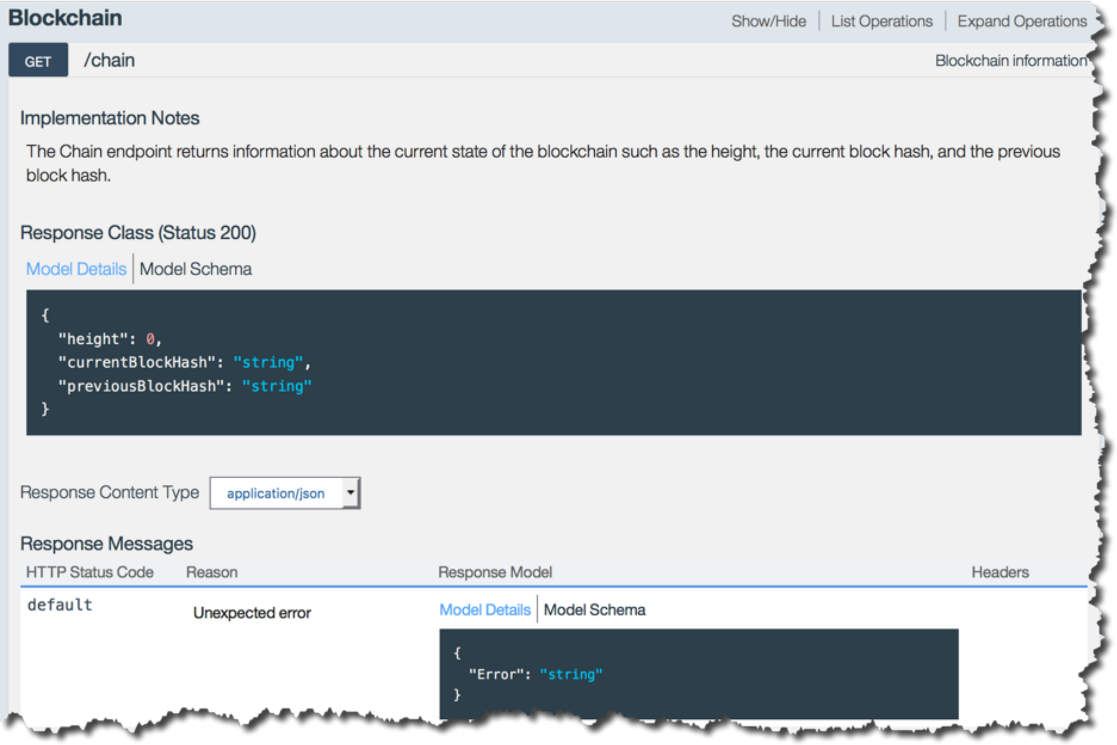
Blockchain

GET /chain

This reveals the **GET /chain** operation which is a valid method to call on the peer.

__3. Click 'Expand Operations' to view information about this API.

This reveals the input and output data formats.



Blockchain
Show/Hide | List Operations | Expand Operations
GET
/chain
Blockchain information

Implementation Notes

The Chain endpoint returns information about the current state of the blockchain such as the height, the current block hash, and the previous block hash.

Response Class (Status 200)

Model Details | Model Schema

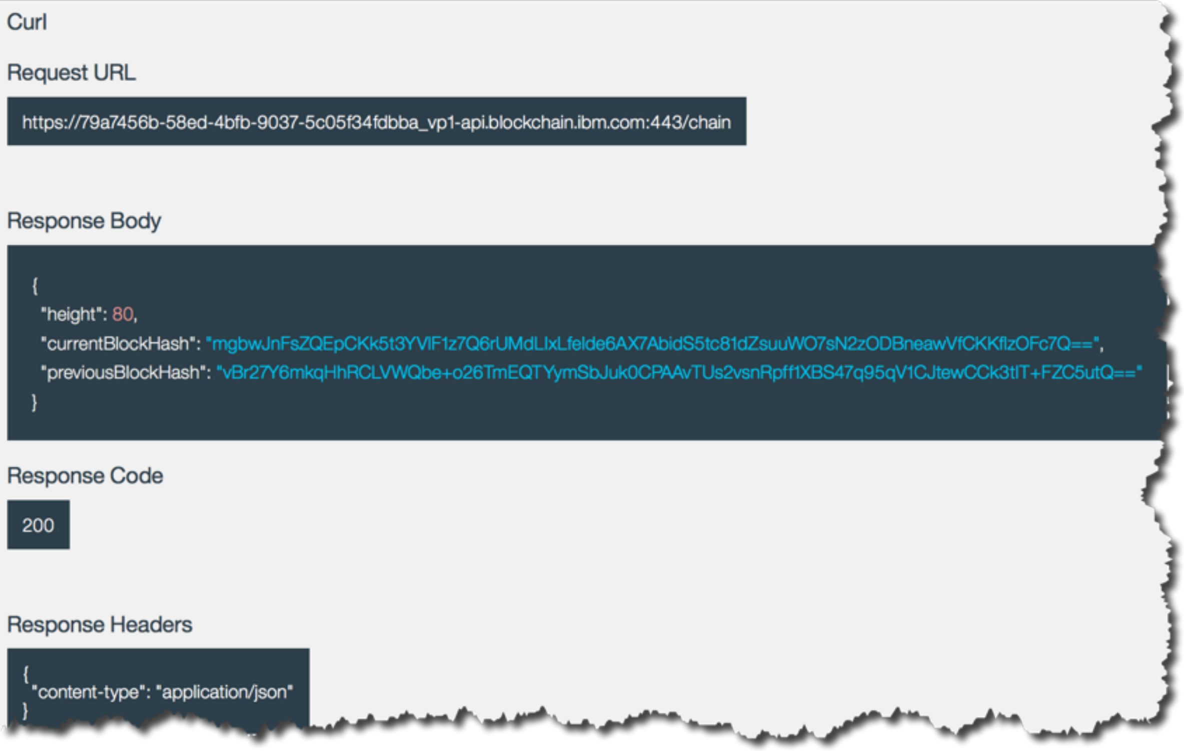
```
{
  "height": 0,
  "currentBlockHash": "string",
  "previousBlockHash": "string"
}
```

Response Content Type: application/json

Response Messages

HTTP Status Code	Reason	Response Model	Headers
default	Unexpected error	Model Details | Model Schema { "Error": "string" }	

__4. Click ‘ Try It Out’ to invoke the API.



Curl

Request URL

```
https://79a7456b-58ed-4bfb-9037-5c05f34fdbba_vp1-api.blockchain.ibm.com:443/chain
```

Response Body

```
{
  "height": 80,
  "currentBlockHash": "mgbwJnFsZQEpCKk5t3YVlF1z7Q6rUMdLlxLfelde6AX7AbidS5tc81dZsuuWO7sN2zODBneawVfCKKflzOFc7Q==",
  "previousBlockHash": "vBr27Y6mkqHhRCLVWQbe+o26TmEQTYymSbJuk0CPAAvTUs2vsnRpff1XBS47q95qV1CJtewCCk3tlT+FZC5utQ=="
}
```

Response Code

```
200
```

Response Headers

```
{
 "content-type": "application/json"
}
```

Review the displayed fields:

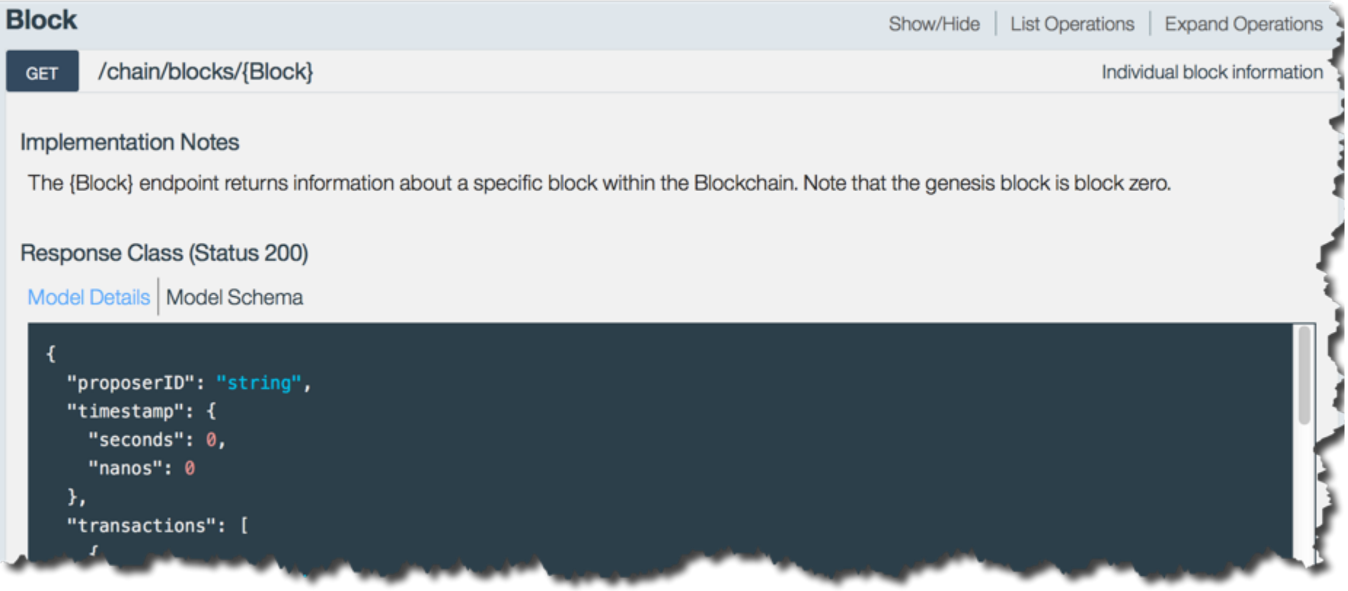
- The *Request URL* shows the URL that was invoked, including the endpoint information of the peer (hostname:port) and the method call (/chain).

- The *Response Body* shows the information that was returned including, importantly, the height of the blockchain.

- The *Response Code* 200 shows that the request was successful.

- The *Response Headers* confirms that the response body has been returned in a JSON data structure.

__5. Expand the ‘Block’ section and review the information on how to interrogate an individual block in the Blockchain.



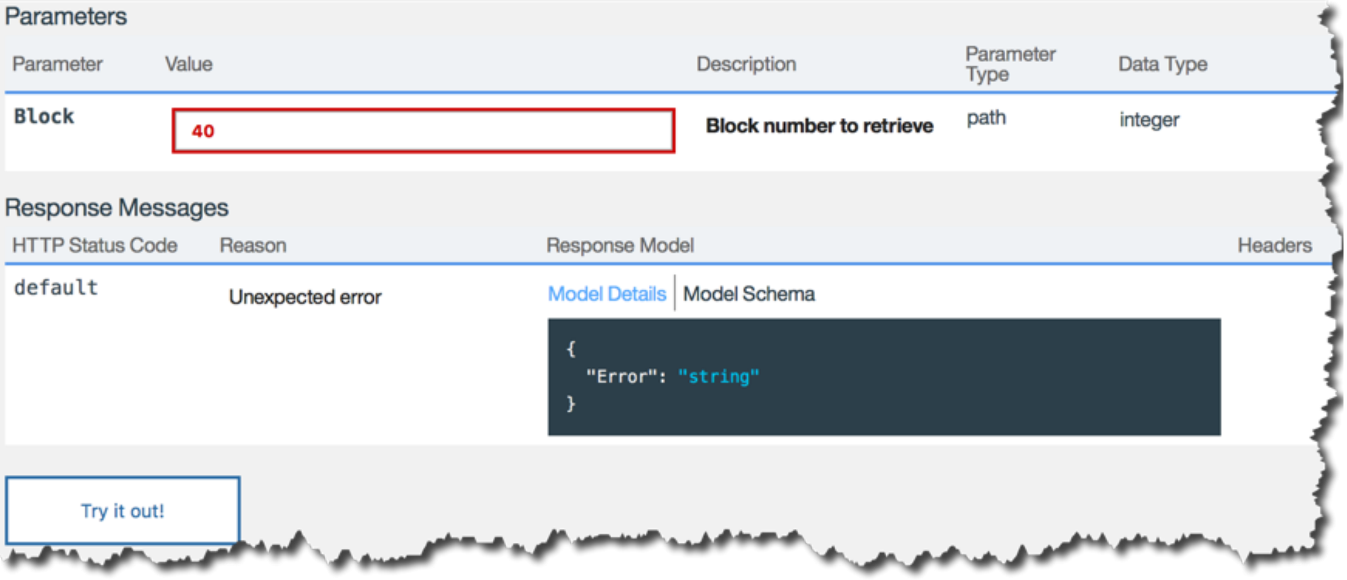
Implementation Notes

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

Response Class (Status 200)

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

__6. Fill in the ‘Block’ parameter to be a number less than the height of the chain and click ‘Try it out!’.



Parameters

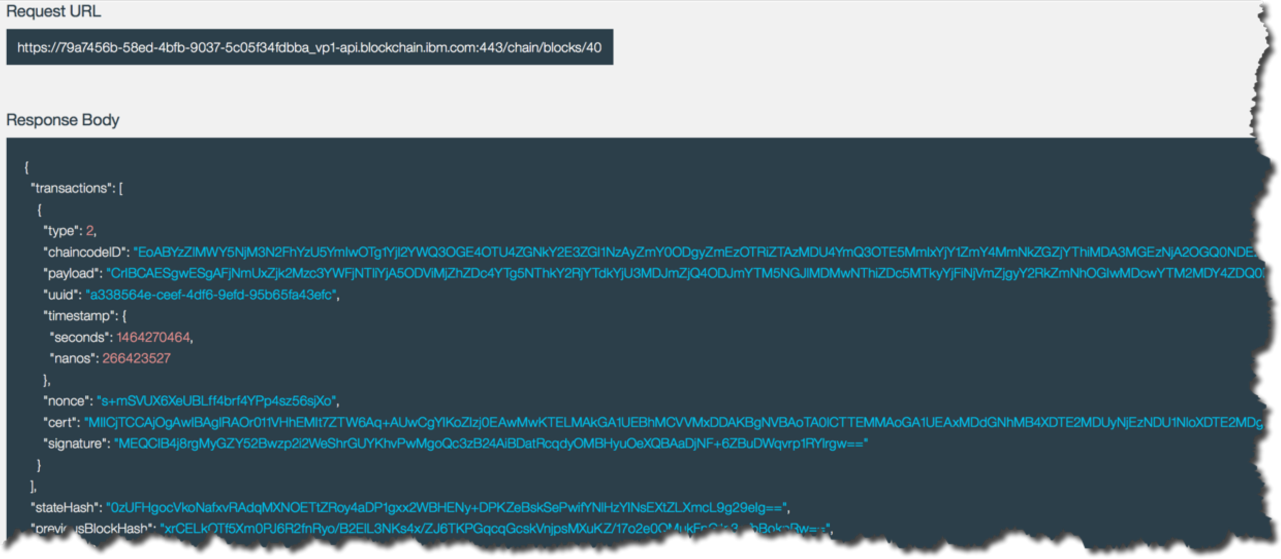
Parameter	Value	Description	Parameter Type	Data Type
Block	40	Block number to retrieve	path	integer

Response Messages

HTTP Status Code	Reason	Response Model	Headers

default	Unexpected error	Model Details | Model Schema
		{
		  "Error": "string"
		}	

__7. Review the information returned in the Response Body.



Request URL

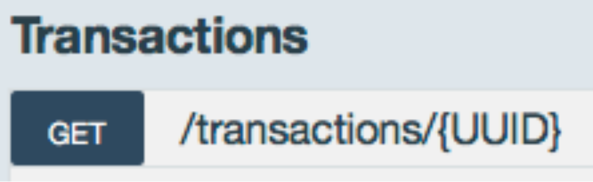
```
https://79a7456b-58ed-4bfb-9037-5c05f34fdbba_vp1-api.blockchain.ibm.com:443/chain/blocks/40
```

Response Body

```
{
 "transactions": [
  {
   "type": 2,
   "chaincodeID": "EoABYzZIMWY5NjM3N2FhYzU5YmIwOTg1YjI2YWQ3OGE4OTU4ZGNkY2E3ZGI1NzAyZmY0ODgyZmEzOTRiZTAzMDU4YmQ3OTE5MmIxYjY1ZmY4MmNkZGZjYThiMDA3MGEzNjA2OGQ0NDE
   "payload": "CrIBCAESgwESgAFjNmUxZjk2Mzc3YWFjNTliYjA5ODViMjZhZDc4YTg5NThkY2RjYTdkYjU3MDJmZjQ4ODJmYTM5NGJlMDMwNThiZDc5MTkyYjFiNjVmZjgyY2RkZmNhOGIwMDcwYTM2MDY4ZDQ0
   "uuid": "a338564e-ceef-4df6-9efd-95b65fa43efc",
   "timestamp": {
    "seconds": 1464270464,
    "nanos": 266423527
   },
   "nonce": "s+mSVUX6XeUBLff4brf4YPp4sz56sjXo",
   "cert": "MIICjTCCAjOgAwIBAgIRAOr011VHhEMIt7ZTW6Aq+AUwCgYIKoZIzj0EAwMwKTELMAkGA1UEBhMCVVMxDDAKBgNVBAoTA0lCTTEMMAoGA1UEAxMDdGNhMB4XDTE2MDUyNjEzNDU1NloXDTE2MDg
   "signature": "MEQCIB4j8rgMyGZY52Bwzp2i2WeShrGUYKhvPwMgoQc3zB24AiBDatRcqdyOMBHyuOeXQBAaDjNF+6ZBuDWqvrp1RYIrgw=="
  }
 ],
 "stateHash": "0zUFHgocVkoNafxvRAdqMXNOETtZRoy4aDP1gxx2WBHENy+DPKZeBskSePwifYNIHzYINsEXtZLXmcL9g29eIg==",
 "previousBlockHash": "xrCELkOTf5Xm0PJ6R2fnRyo/B2EIL3NKs4x/ZJ6TKPGqcqGcskVnjpsMXuKZ/17o2e0QMukFr bBokRw==",
```

__8. Copy the UUID field of a transaction from a block; this will be of the form “a338564e-ceef-4df6-9efd-95b65fa43efc”.

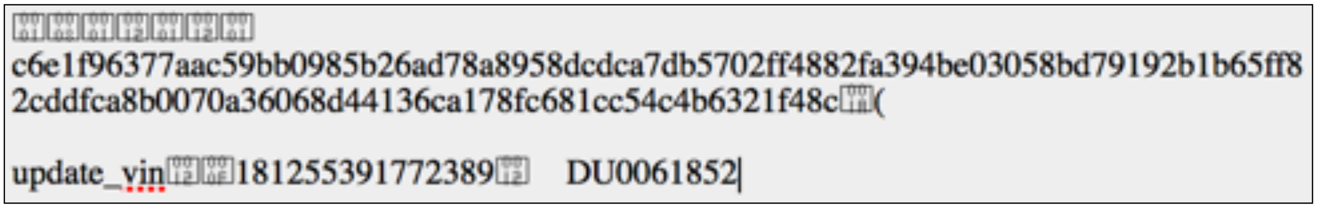
__9. Click the ‘Transactions’ section.



This reveals the **GET /transactions/{UUID}** operation which is a valid method to call on the peer.

__10. Paste the transaction UUID and click ‘Try it out!’.

The ‘payload’ field is base64 encoded (use a web tool such as http://www.base64decode.org for decoding this information); when decoded you’ll see that the payload includes the chaincode ID of the smart contract being called together with its input parameters. For example:



```
c6e1f96377aac59bb0985b26ad78a8958dcdca7db5702ff4882fa394be03058bd79192b1b65ff8
2cddfca8b0070a36068d44136ca178fc681cc54c4b6321f48c(

update_vin181255391772389    DU0061852
```

Note that this application does not encrypt the transactions, so the payloads are visible (albeit base64 encoded) to all.

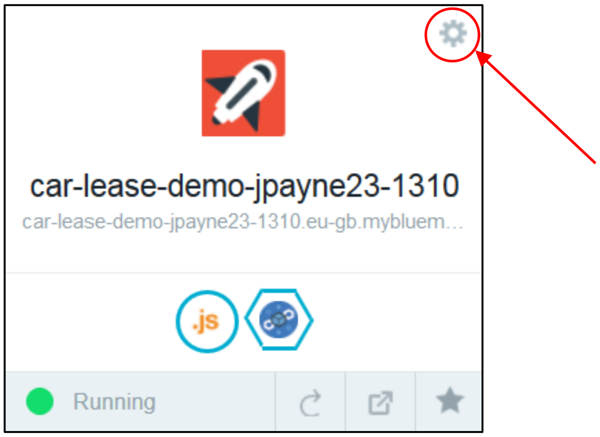
__11. Now spend some time interacting with the other APIs available to you.

Section 3. Removing the sample application

The final section of this lab aims to stop and remove the Blockchain service you created.

__1. Return to the Bluemix Dashboard by clicking DASHBOARD.

__2. Click the Settings icon in the car lease demo application.

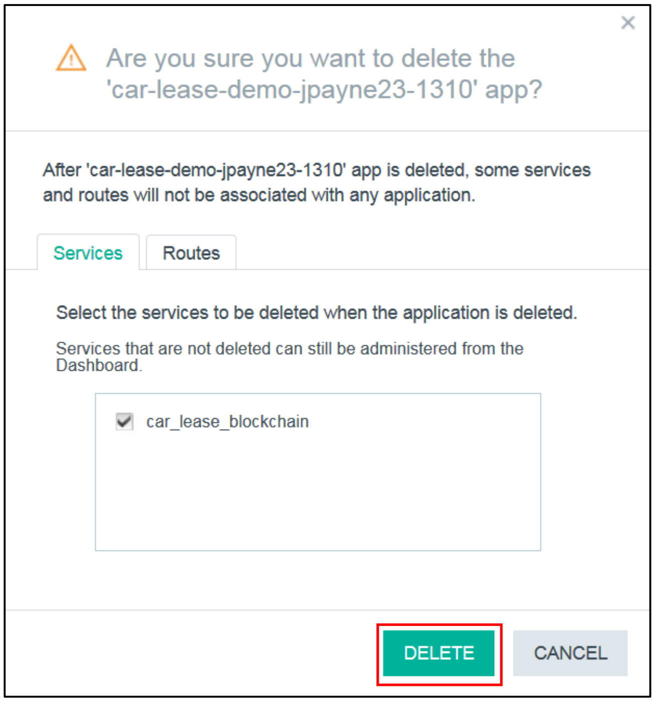


car-lease-demo-jpayne23-1310

car-lease-demo-jpayne23-1310.eu-gb.mybluem...

__3. Select 'Delete App' from the menu.

__4. Ensure that the ‘car_lease_blockchain’ service is also selected for deletion and click ‘Delete’.



⚠ Are you sure you want to delete the 'car-lease-demo-jpayne23-1310' app?

After 'car-lease-demo-jpayne23-1310' app is deleted, some services and routes will not be associated with any application.

Select the services to be deleted when the application is deleted.

Services that are not deleted can still be administered from the Dashboard.

☑ car_lease_blockchain

__5. Wait for the items to be stopped and deleted. Once this is done, both the application and the associated service will no longer be visible in the Bluemix dashboard.

Congratulations on completing Lab two – “Blockchain Explored”!

Appendix A. 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 B. 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	
System z	*Tivoli*	*WebSphere*	*Workplace*	*System p*	

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

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.

Please Recycle