



IBM **Bluemix**TM

Hands-on Exercise

Gather Internet of Things sensor data using Bluemix and Node-RED.

Contents

Introduction	3
Step 0. Visualizing Mobile Sensor Data using IBM Internet of Things Foundation	4
Step 1. Create Your Node-RED Application	6
Step 2. Create your Node-RED flow.....	9
Step 3. Adding further functionality to your Node-RED flow.....	11
Step 4. Using Your Phone's Data to Send Twitter Messages.....	13
Step 5. Use Twitter Data as Input for a Sentiment Analysis	16
Step 6. Build your own User Interface.....	19
Step 7. Protect your Node-RED flow with Username/Password (Optional)	25

Introduction

This exercise will show you how to deploy an application from the IBM Bluemix Web User Interface. We will create a Node-RED application running on SDK for Node.js runtime. Bluemix and Node-RED will be used to collect sensor data from a mobile device. We will then set up the application to send messages to Twitter based on the real time sensor information from the mobile device. To end things off, we will use IoT foundation to create a real-time sentiment analysis of twitter feeds.

About IBM Bluemix

Bluemix is an open-standard, cloud-based platform for building, managing, and running applications of all types (web, mobile, big data, new smart devices, and so on).

- The developer can choose any language runtime or bring their own. Zero to production in one command.
- A catalog of IBM, third party, and open source API services allow the developer to stitch an application together in minutes.
- Development, monitoring, deployment, and logging tools allow the developer to run the entire application.
- Sign up in minutes. Pay as you go and subscription models offer choice and flexibility.

About Node-RED

Node-RED is a visual tool for wiring the internet of things – connecting hardware devices, APIs and online services in a new and interesting way. Node-RED provides a browser-based flow editor that makes it easy to wire together flows using the wide range nodes in the palette. Flows can be then deployed to the runtime in a single-click.

- JavaScript functions can be created within the editor using a rich text editor.
- A built-in library allows you to save useful functions, templates or flows for re-use.
- See <https://nodered.org> for more information.

Prerequisites

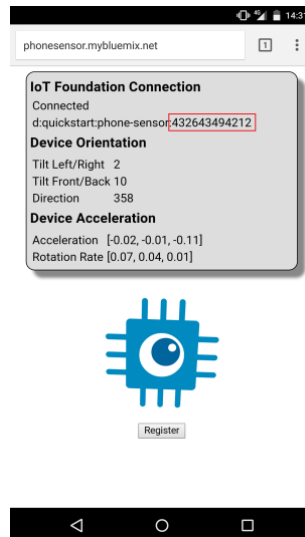
- Register on IBM Bluemix at <https://bluemix.net>

Step 0. Visualizing Mobile Sensor Data using IBM Internet of Things Foundation

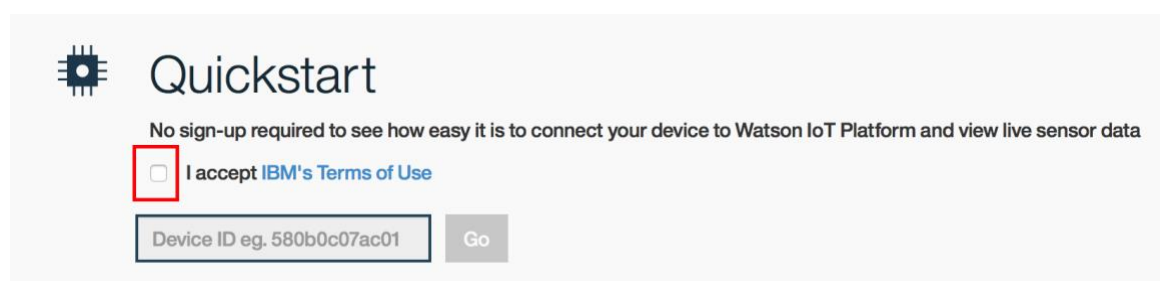
1. Before we begin with the main lab (starting in Step 1), we will visualize sensor data from a mobile device using IBM Internet of Things foundation quick start. We will then create our own Bluemix application to import sensor data and show how you can act on the data in real-time.
2. We will first set up your mobile device to send its sensor data to the IBM Internet of Things Foundation. Open a browser on your mobile device and go to <https://phonesensor.mybluemix.net>.

Note: If your phone shows “Disconnected” try: (removed the ‘s’ in https) <http://phonesensor.mybluemix.net/>.

3. Copy or write down the set of numbers presented after “d:quickstart:phone-sensor:”. This will be your device ID and will be needed in the next step.



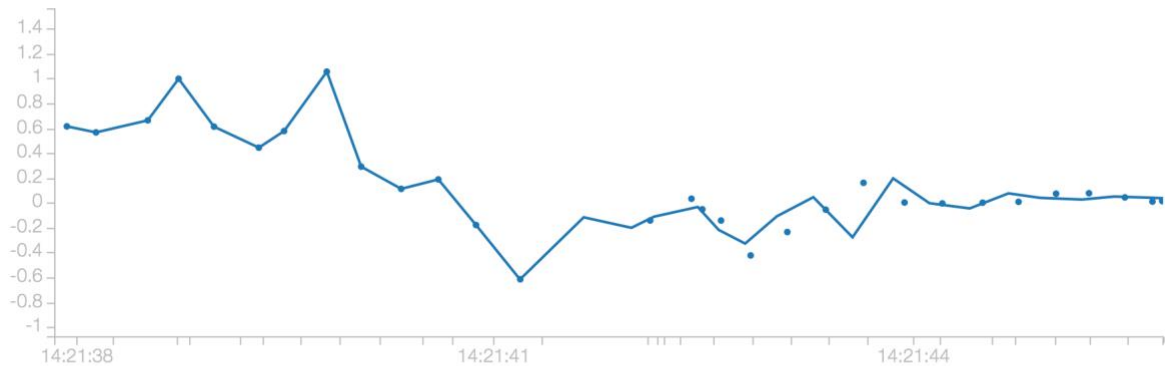
4. Open a new browser on your computer and navigate to <https://quickstart.internetofthings.ibmcloud.com/#/>
5. Accept the terms and enter the set of numbers you wrote down then click on ‘GO’.



6. Make sure you keep the browser with <https://phonesensor.mybluemix.net> open on your phone and watch the sensor data flow in the IBM Internet of Things Foundation quick start window. Try moving your phone and selecting the different phone sensors.

If you lock your phone the data flow will stop. Keep in mind that we are not installing anything into your phone, just reading the built-in sensors, which also means that the ID will change after some time.

Refresh the web to get a new ID.



Event	Datapoint	Value	Time Received
status	accelX	0.07	Mar 29, 2017 2:21:46 PM
status	accelY	-0.01	Mar 29, 2017 2:21:46 PM

Step 1. Create Your Node-RED Application

1. In a browser navigate to <https://bluemix.net>
2. Select 'LOG IN' then enter your log in information and press 'SIGN IN'. You should be seeing your dashboard view:

The screenshot shows the IBM Bluemix Apps dashboard. At the top, there's a navigation bar with 'Docs', 'IBM Bluemix Apps', a user profile 'Tea Suuronen's Account', and a '262' badge. Below the navigation bar is a search bar and a 'Create Application' button. The main content area is divided into two sections: 'All Applications (2)' and 'All Services (8)'. The 'All Applications (2)' section shows a table of Cloud Foundry Applications with columns: NAME, ROUTE, MEMORY (MB), INSTANCES, RUNNING, STATE, and ACTIONS. There are two applications listed: 'Diipadaaba' and 'TeaWatsonHarjoitteet', both with a state of 'Stopped'. The 'All Services (8)' section shows a table of Services with columns: NAME, SERVICE OFFERING, PLAN, and ACTIONS. There are no services listed in this section.

NAME	ROUTE	MEMORY (MB)	INSTANCES	RUNNING	STATE	ACTIONS
Diipadaaba	Diipadaaba.mybluemix.net	512	1	0	Stopped	[Icon]
TeaWatsonHarjoitteet	TeaWatsonHarjoitteet.mybluemix.net	512	1	0	Stopped	[Icon]

NAME	SERVICE OFFERING	PLAN	ACTIONS
------	------------------	------	---------

3. Select the 'CATALOG' view.
4. Locate the **NODE-RED Starter** in the boilerplate section (Apps → Boilerplates) of the catalog and click on it.

The image shows a 'Node-RED Starter' boilerplate card. It features a red circular icon with a white Node-RED logo. The text reads: 'Node-RED Starter', 'This application demonstrates how to run the Node-RED open-source project within IBM Bluemix.', and a green 'Community' badge.

5. Enter a name for your application, as shown below (host will automatically be completed). The host name must be unique on Bluemix, so please choose a name with your company name or initials to try to make a unique name. Press 'CREATE'.

Create a Cloud Foundry Application

Node-RED Starter

This application demonstrates how to run the Node-RED open-source project within IBM Bluemix.

[Community](#)

[View Docs](#)

VERSION	0.5.0
TYPE	Boilerplate
REGION	US South

App name:

EP-Testi-phone

Host name:

EP-Testi-phone

Domain:

mybluemix.net

Selected Plan:

SDK for Node.js™

Default

Cloudant NoSQL DB

Lite

Need Help? [Contact Bluemix Sales](#)

Estimate Monthly Cost [Cost Calculator](#)

Create

- Your application is now staging and will be up and running in a short while. Press **'OVERVIEW'** to see information about your application.

Note: You may be given green light few seconds before the application has finished staging. If you want, check the logs to see if the application is still being built.

IBM Bluemix Cloud Foundry Applications

262

← Dashboard

Getting Started

Overview

Runtime

Connections

Logs

Monitoring

EP-Testi-phone Status: Your app is starting

Start coding with Node-RED

Last updated: 22 August 2015

- After your application has started, click on the **Routes URL** or enter the following URL in a browser:


```
http://<yourhost>.mybluemix.net
```

The Node-RED landing page displays.
- Click **Go to your Node-RED flow editor**. This opens up a browser-based flow editor that makes it easy to wire together devices, APIs, and online services by using the wide range of nodes included in its palette.

Customizing your Node-RED instance

Before you begin, install the Cloud Foundry command line interface.

[Download Cloud Foundry command line interface](#)

- When fully staged click on the URL of your application, this launches the Node-RED main page.

iotsensors-sc

Running

iotsensors-sc.mybluemix.net

Routes

↺

⌂

⋮

8. Now click **Go to your Node-RED flow editor** to open the flow editor.



Go to your Node-RED flow editor

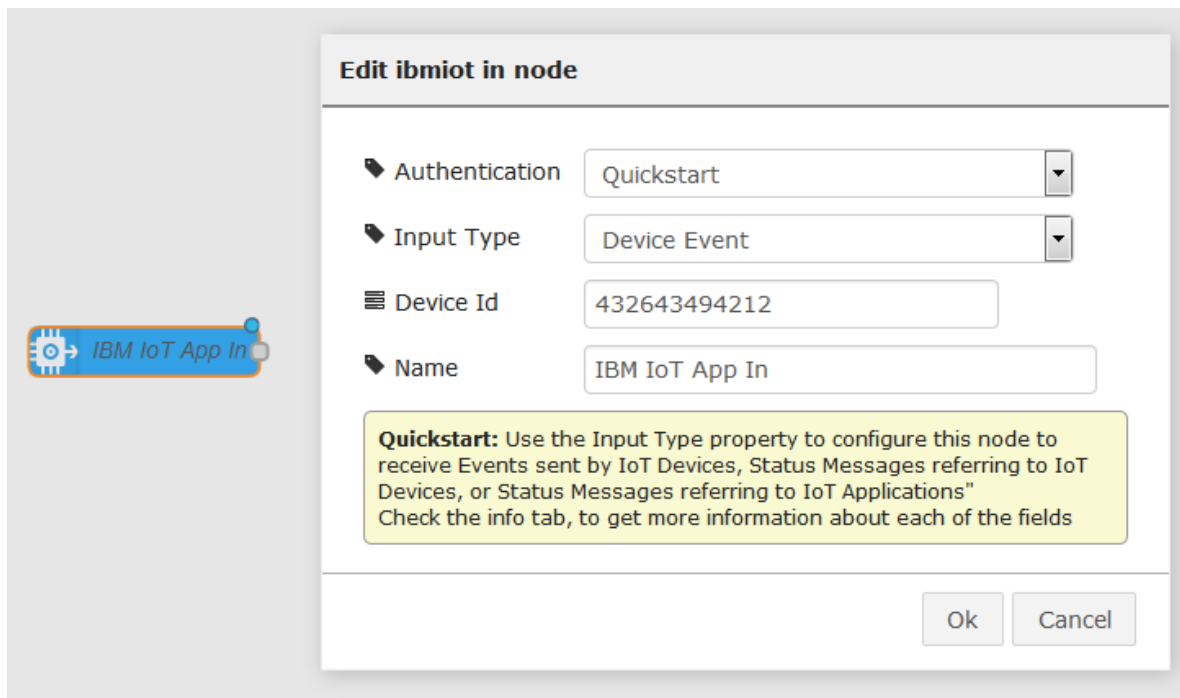
9. When using Node-RED we build our apps using this graphical editor interface to wire together the blocks we need. We can simply drag and drop the blocks from the left menu into the workspace in the center of the screen and connect them to create a new flow.

Step 2. Create your Node-RED flow.

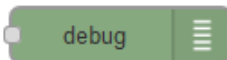
1. Locate the ibmiot node under the input section in the palette window.



2. Drag the node onto the workspace in the middle of the screen.
3. Double click on your new ibmiot node and select 'QUICKSTART' under 'AUTHENTICATION'.
4. Enter the set of numbers you wrote down / copied in Step 0. or Step 2. in 'DEVICE ID'. Press 'OK'.



5. Locate the debug node under the output section in the palette window.



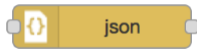
6. Drag the node onto the workspace.
7. Connect the ibmiot node with the debug node by dragging the output and input dots together.



If you see a blue dot in the right corner of the node, it means that you have made changes that have not been deploy to the application yet.

8. Click 'DEPLOY' in the upper right corner.
9. Click on the "debug" tab on the right of the screen to see the phone sensor data flowing. Make sure you have the phone sensor browser tab open on your phone. The web application is only sending sensor data to Bluemix as long as the browser is kept open and the screen is on. Try moving your phone up/down or sideways to see the sensor data changing in real time.

The data will be shown in a compact mode in the debug tag. In order to get 'json' format, add the json node from the palette under the function section.



Drag and drop the node between the IoT node and the debug node, as shown in the next picture:



Before json node:

```
29/03/2017, 14.57.29  node: ff497581.8c0d68
iot-2/type/phone-sensor/id/490788535974/evt/status/fmt/json :
msg.payload : Object
▼ object
  ▼ d: object
    accelX: 0
    accelY: 0
    accelZ: 0.03
    rotationX: 0.04
    rotationY: -0.13
    rotationZ: 0.06
    tiltLR: 2
    tiltFB: -1
    direction: 1
```

```
29/03/2017, 14.57.29  node: ff497581.8c0d68
iot-2/type/phone-sensor/id/490788535974/evt/status/fmt/json :
msg.payload : Object
▶ { d: object }
```

After json node:

```
29/03/2017, 14.55.56  node: ff497581.8c0d68
iot-2/type/phone-sensor/id/490788535974/evt/status/fmt/json :
msg.payload : string[141]
{"d":
{"accelX":0.04,"accelY":0.12,"accelZ":0.1
6.12,"tiltLR":-
7,"tiltFB":30,"direction":347}}"
```

```
29/03/2017, 14.55.56  node: ff497581.8c0d68
iot-2/type/phone-sensor/id/490788535974/evt/status/fmt/json :
msg.payload : string[140]
{"d":{"accelX":-
0.07,"accelY":0,"accelZ":0.02,"rotationX"
3.62,"rotationY":-
0.44,"rotationZ":1.41,"tiltLR":-
7,"tiltFB":31,"direction":346}}"
```

Step 3. Adding further functionality to your Node-RED flow.

1. Delete the connection between the ibmiot node and the debug node.
2. Add a function node under function in the palette window between the ibmiot node and the debug node.
3. In the function node, add the following code:

```
if (parseFloat(msg.payload.d.tiltFB) > 45) {  
  msg.payload = "Phone rotated more than 45 degrees";  
} else return;  
return msg;
```

This code will create the output “Phone rotated more than 45 degrees” in the debug tab if your phone is rotated more than 45 degrees. tiltFB is the tilt in the front-back direction of the phone.

Edit function node

Name

Name

Function

```
1 if (parseFloat(msg.payload.d.tiltFB) > 45) {  
2   msg.payload = "Phone rotated more than 45 degrees";  
3 } else return;  
4 return msg;  
5
```

Outputs

1

See the Info tab for help writing functions.

Ok

Cancel

4. Connect the ibmiot node to the function node, and the function node to the debug node.



5. Click 'DEPLOY' and try rotating your phone in the front-back direction. Make sure the <https://phonesensor.mybluemix.net> browser is open. See what happens in the debug tab when the phone is rotated more than 45 degrees.

6. You can try different functions and outputs by editing the function node. The different sensors are sent through the following payloads:

- `msg.payload.d.accelX`
- `msg.payload.d.accelY`
- `msg.payload.d.accelZ`
- `msg.payload.d.rotationX`
- `msg.payload.d.rotationY`
- `msg.payload.d.rotationZ`
- `msg.payload.d.tiltFB`
- `msg.payload.d.tiltLR`
- `msg.payload.d.direction`

Step 4. Using Your Phone's Data to Send Twitter Messages

1. Node-RED can be set up to send Twitter messages based on the incoming sensor data.
2. Go to <https://twitter.com> and make sure you are signed into your Twitter account. If you do not have a Twitter account, go to <https://twitter.com> and sign up for free.
3. In your Node-RED window, 'DELETE' the debug node.
4. Add a delay node found under functions in the palette window. We need this node to make sure only one Tweet is posted. Without this node, the same message would be posted over and over again on Twitter.
5. Double click on the delay node and change the 'ACTION' to 'Limit rate to' and the 'RATE' to '1 msg(s) per Day'.

Edit delay node

Action

Limit rate to

Rate

1

msg(s) per

Day

☐ drop intermediate messages

Name

Name

Ok

Cancel

6. Finally, add a Twitter out node found under social in the palette window.
7. Double click on the Twitter out node and click on the edit button next to 'Add new twitter-credentials...'.

Edit twitter out node

Twitter

Add new twitter-credentials...

Name

Tweet

Ok

Cancel

8. Click on 'Click here to authenticate with Twitter.'

Add new twitter-credentials config node

Click here to authenticate with Twitter.

Add

Cancel

9. This will open a new browser tab, click on 'Authorize app'. You should receive the message 'Authorised - you can close this window and return to Node-RED'. Close the browser tab.

Authorize Node RED to use your account?

Authorize app

Cancel



Node RED

nodered.org

Node-RED Twitter node

This application will be able to:


- Read Tweets from your timeline.
- See who you follow, and follow new people.
- Update your profile.
- Post Tweets for you.
- Access your direct messages.

Will not be able to:

- See your Twitter password.

10. Go back to your Node-RED tab. You should now see your Twitter username next to the Twitter ID. Click 'ADD'. You are now done setting up the connection to Twitter. Click 'OK'.

Add new twitter-credentials config node

 Twitter ID

Add

Cancel

11. Connect all your nodes, it should look like this:



12. When you rotate your phone (with the phone sensor browser tab open) more than 45 degrees it will post a message to Twitter.

Step 5. Use Twitter Data as Input for a Sentiment Analysis

1. We will now create a flow that instead of posting to Twitter based on input data, we gather data *from* twitter, analyze the sentiment of the data and send the data through the IoT foundation to a live graph.
2. 'DELETE' your previous flow.
3. Create the following flow:



1. Twitter node found under 'Social'.
 2. Sentiment node found under 'Analysis'.
 3. Function node found under 'function'.
 4. IoT Out node found under 'output'.
4. Click on the Twitter node. If needed, add your credentials like done in the previous step. Configure the node like the picture below, where 'for' can be any key word that gets a lot of twitter posts (for example IBM or Google).

Edit twitter in node

Twitter ID

@atterstromvic

Search

all public tweets

for

ibm

Name

Name

Tip: Use commas without spaces between multiple search terms.
Comma = OR, Space = AND.
The Twitter API WILL NOT deliver 100% of all tweets.
Tweets of who you follow will include their retweets and favourites.

Ok

Cancel

5. Add the following code to the function node:

//This code creates a JSON array based on the data from the previous node. We extract the relevant information, like the tweet, the sentiment score, the location of the tweet and the number of followers.

```
msg.payload = JSON.stringify({
  d: {
    Description: "All tweets with a connection to your keyword",
    Sentiment: msg.sentiment.score,
    Tweet: msg.payload,
    Location: msg.location,
    Followers: msg.tweet.user.followers_count
  }
});
return msg;
```

6. Configure the IoT Out node like the picture below, where 'Device Id' is a unique name of your choice, for example your first name. This name is not linked to the device id used in the previous exercise. The value in 'Data' is not important, as this node is programmed so that any information sent from the previous node in msg.payload will override the value of 'Data'.

Edit ibmiot out node

Authentication

Quickstart

Device Id

Outputadress

Event Type

Sentiment

☒ Data

override

Name

IBM IoT App Out

Note: If there is a property in the message that corresponds to any of the values entered above, then the property in the message takes precedence. See the Info tab for more details.
Example JSON device event: {"d":{"myName":"Arduino Uno", "temperature":989}}

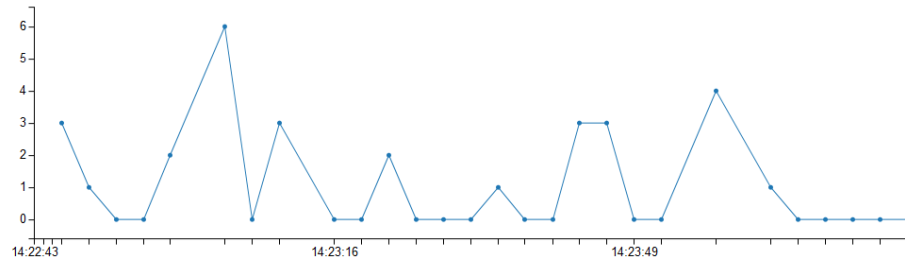
Ok

Cancel

7. Connect your flow together:



8. Go to <https://quickstart.internetofthings.ibmcloud.com/> and enter your unique device ID, entered in step 6. You should now be seeing a graph with a live flow of the sentiment analysis of your chosen keyword. Anything below 0 is a negative sentiment, 0 is neutral, and anything above 0 is positive.



Event	Datapoint	Value	Time Received
Twitter data	Description	All tweets with a connection to the IBM brand	Sep 3, 2015 2:24:19 PM
Twitter data	Sentiment	0	Sep 3, 2015 2:24:19 PM
Twitter data	Tweet	[ADAGE] How CMOs Are Spending Across The Customer-Buying Cycle: IBM and the CMO Club." image="http://%%img_ser... http://t.co/Z0tfexsluB	Sep 3, 2015 2:24:19 PM
Twitter data	Location.place	New York	Sep 3, 2015 2:24:19 PM
Twitter data	Followers	6906	Sep 3, 2015 2:24:19 PM