{ Lab }

IoT application in IBM Cloud with Node-RED and IBM Watson

June 2019 – v1.1

Authors: B. Marolleau, C. Bacle, C. Lalevée, Y. Holvoet
June 2019 – v1.1

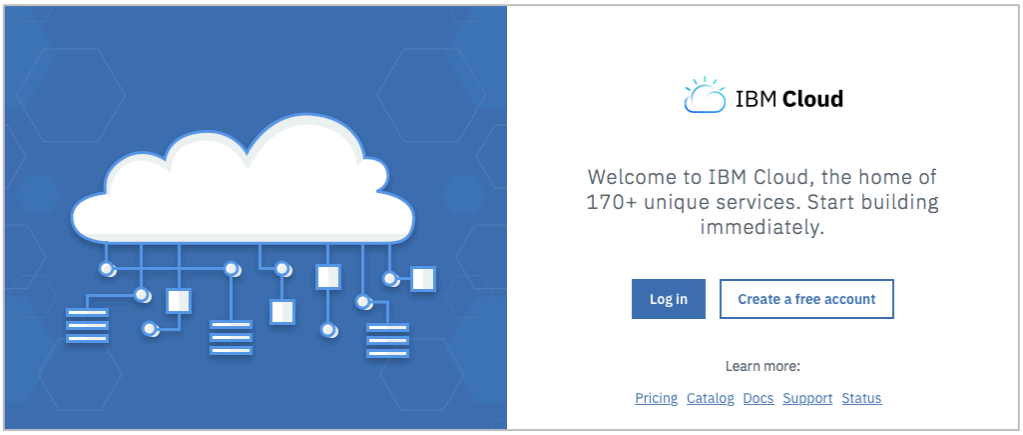
Agenda

Before Starting

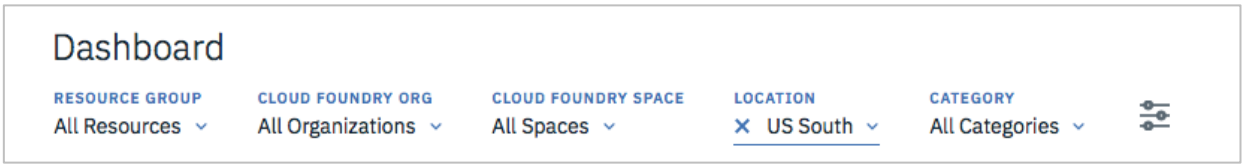
Information

This hands-on required to have an IBM Cloud account. If you don't, you can create one here: http://bluemix.net/.
A complete guideline can also be found on the "IBM Cloud & Watson Day" web page here: http://ibmcloud-watson-day.mybluemix.net/

- Open a browser and access to IBM Cloud: https://console.bluemix.net.
- If you have an IBM Cloud account, click **Log in**, and enter your IBM ID credentials.
  If you don't have an IBM Cloud account, click **Create a free account**. Enter your email address, and additional information required. You will receive an email with activation link. Once activated, you could use your new free IBM cloud account: log in.



- Select organization, location and space to use during this lab.



- If needed, free resources (GB / #Services) in your IBM Cloud Organization & Spaces to run the lab exercises.
  If you encounter a resource contention (error message saying you are out of resources), clean up your spaces by deleting existing Apps or Services.

Hands-on presentation

Section 1. Overview

In this hands-on session, you will create a Node-RED application in IBM Cloud to collect, store and display virtual sensor data.

Node-RED (https://nodered.org/) is a flow-based programming tool, originally developed by the IBM Emerging Technology Services team (in early 2013) and now a part of JS Foundation.
Traditional development can be very technical, but Node-RED enables you to concentrate on the logic of your workflow and allows fast prototyping.
Node-RED consists of a Node.js-based runtime with a flow editor accessed through a web browser. Within the browser, you create your application by dragging nodes from a customizable palette into a workspace and start to wire them together. With a single click, the application is deployed back to its runtime.

Session objectives are:

- Create & modify an application using Node-RED in IBM Cloud and on a device. In that example, we will use a Raspberry PI 2 B+
- Discover new services & Node-RED to consume or create services (IoT / AI /Database)
- Discover IBM Watson AI and IoT services
- Discover how to integrate Cloud Services with a Raspberry PI or any device using Node-RED and create prototype quickly.

Section 2. Prerequisites

There are 2 kinds of exercises in the present Lab: in the Cloud (IBM Cloud) and local (Node-RED on a Raspberry PI).

1) The necessary hardware (provided);

- Raspberry PI 2 with an internet access and Node-RED installed.
- A PC connected to the Raspberry PI network (provided)

2) Setup & Connectivity check:

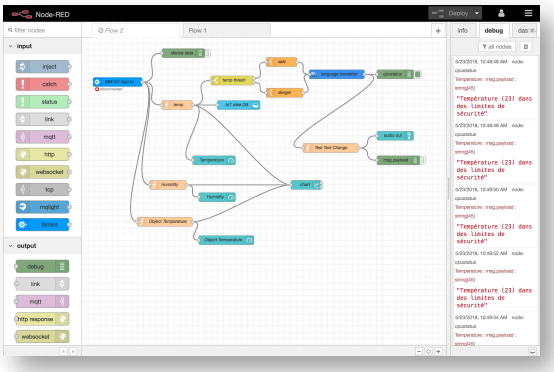
- Power on the Raspberry - USB power supply
- From your PC - go to http://raspberry314-pot.eu-de.mybluemix.net/home .
    - click on your team picture and get the IP of your Raspberry on the local network.

      You will use this address to access your Raspberry from a Browser.
- From your PC, connect to Node-RED using http://YOUR-RASPI-IP:1880 from a Browser (Firefox, …)
- IBM Cloud Sign in / Log on as mentioned in “Before Starting” section above.

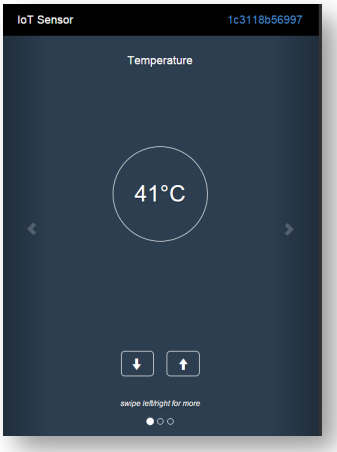
Cloud: Create Node-RED application and Login

Expected results are:

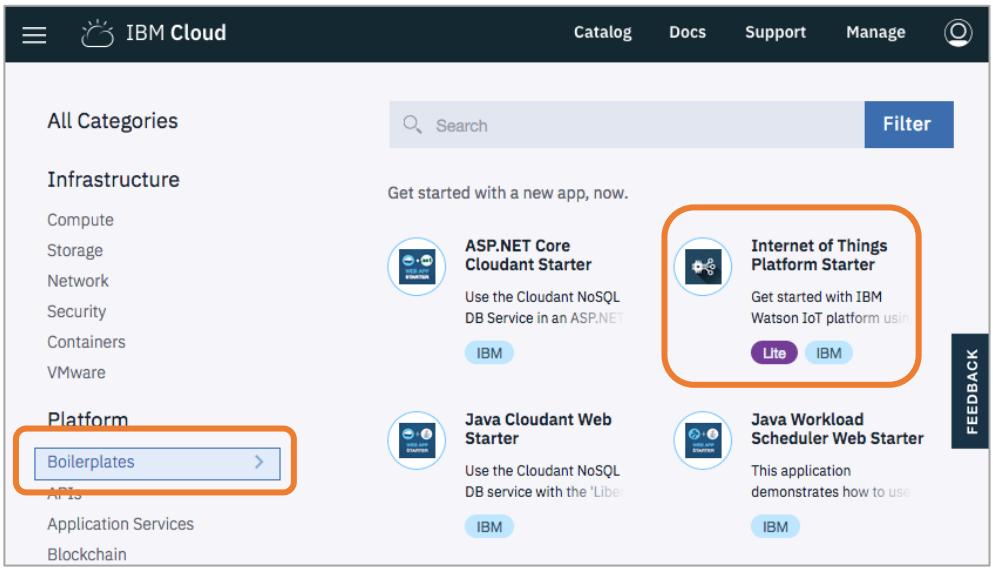
- Your Node-RED application is operational (using Node.js runtime), accessing Cloudant & IoT platform (QuickStart)



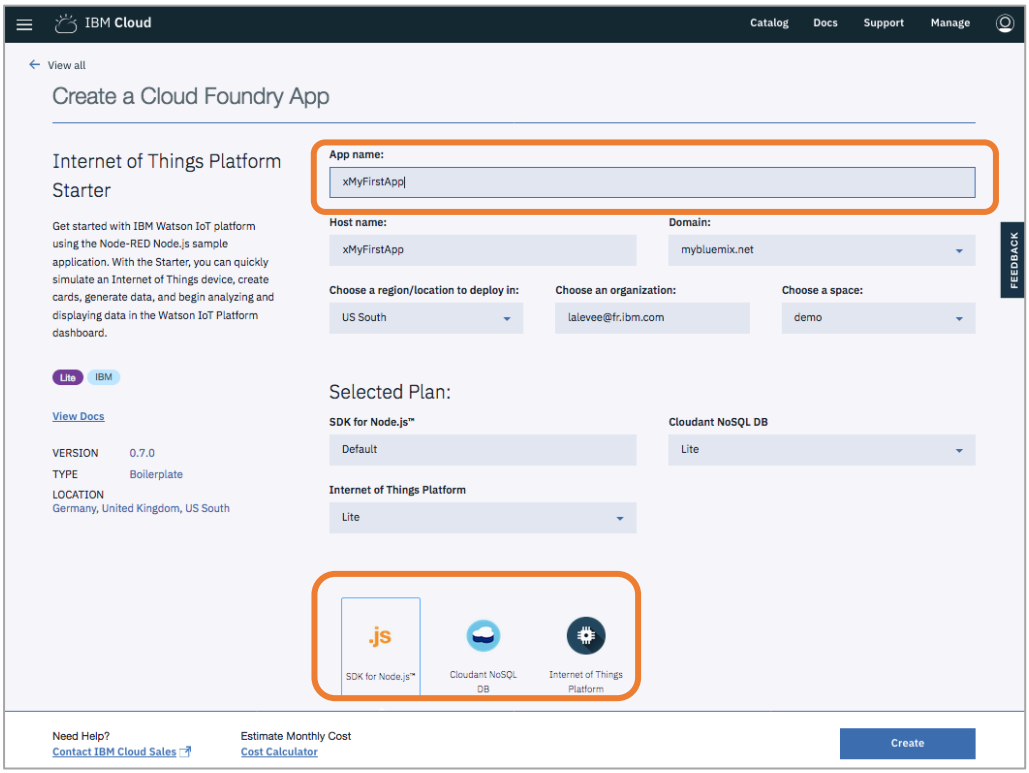
- Your Node-RED app is online (reachable from the Internet), & will be connected to a temperature simulator (sensor)



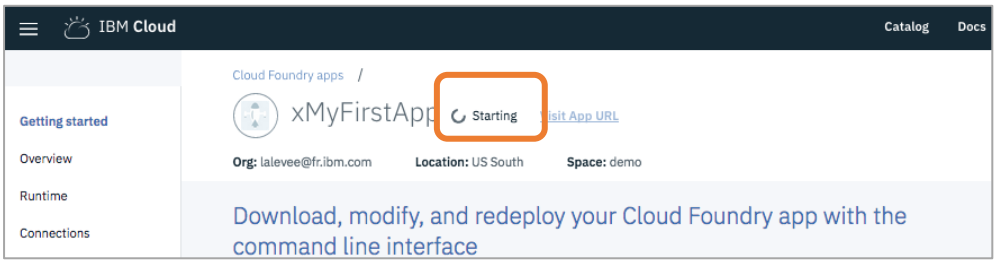
__ 1. In IBM Cloud Catalog, choose **Boilerplate** category



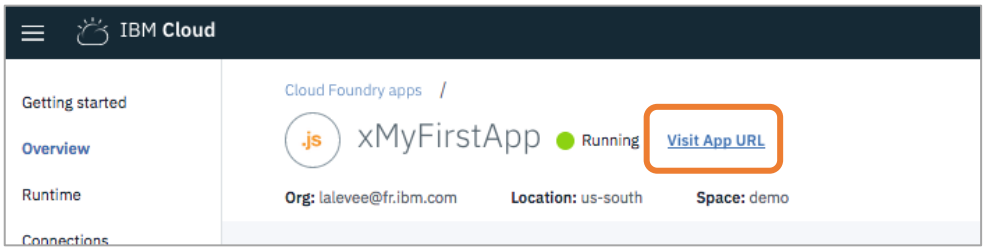
__ 2. Click on **Internet of Things Platform Starter** to create an instance:
Fill in the App Name & Host name fields.
Note: Node-RED is a Node.js based application: using this boilerplate will instantiate a Node.js runtime + a Cloudant (NoSQL DB) service.



__ 3. Click **Create**.
Wait for the environment to be created & the app to start (~4 minutes).

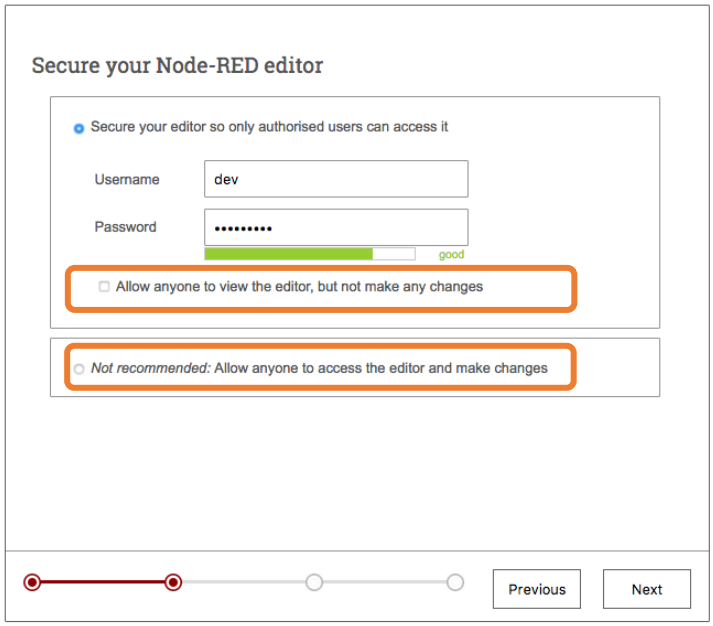


__ 4. By clicking **Visit App URL**, access the Node-RED application

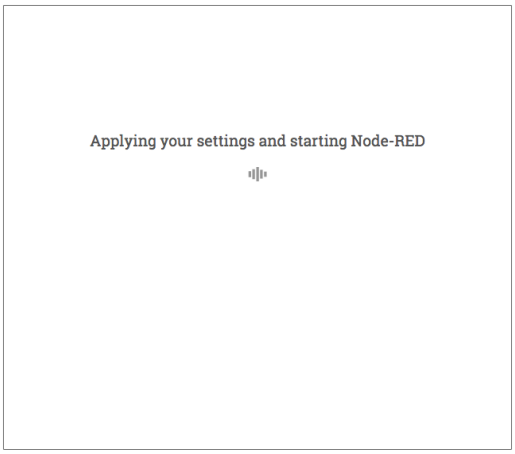


__ 5. Run the wizard to configure authentication: secure your editor with your own credentials so only authorized users can access it (Node-RED has its own authentication system).

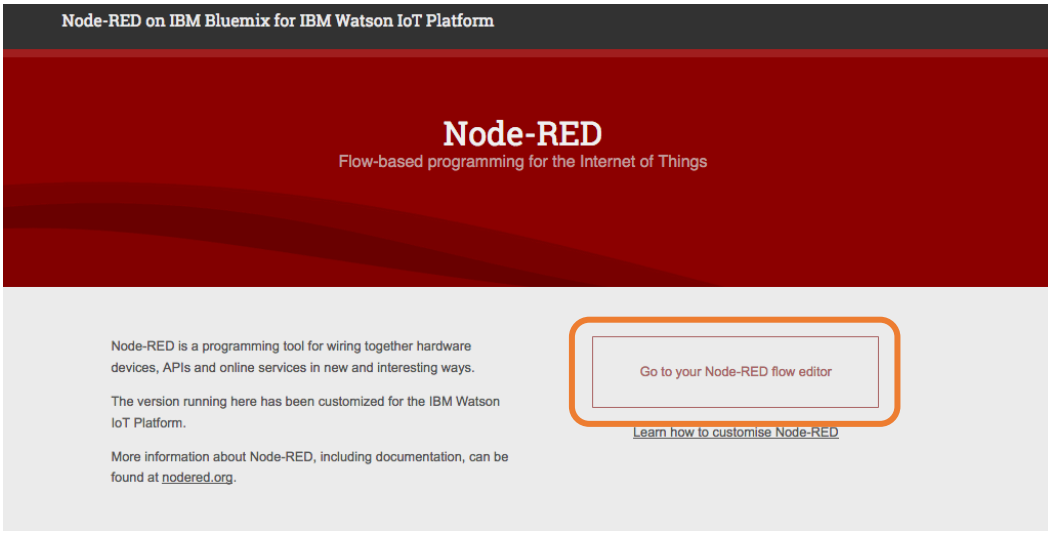
Don't check **Allow anyone to view the editor, but not make any changes** and **Allow anyone to view the editor**.



__ 6. On wizard last step screen, click on **Finish** to start Node-RED.



__ 7. Node-RED is a browser-based editor that makes it easy to wire together flows that can be deployed to the runtime. In the case of IoT, Node-RED is really powerful to quickly test all the possibilities that IBM Cloud offers with different kind of services. Your Node-RED app has a public URL like any web app (you defined it in step 2). Click on **Go to your Node-RED flow editor** and use the credentials provided before.



__ 8. You now have access to the Node-RED UI.
Keep the existing default flow and create a new flow: click **+**. You will use it in next exercise.

1. Cloud: Create a virtual sensor and a new flow

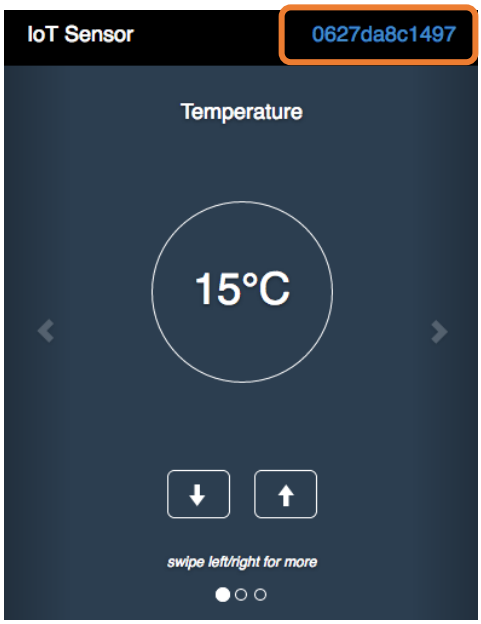
Section 1. Sensors & IoT

__ 1. Open a new window or tab in your browser.

__ 2. To create a sensor simulator, connect to http://ibm.biz/iotsensor
There are 3 simulated sensors:

- Object temperature
- Temperature
- Humidity

The simulator (from IBM Cloud IoT Quickstart) connects automatically and starts publishing data.
It must remain connected to visualize the data.
Use the simulator buttons to change the simulated sensor readings. Data is published periodically.



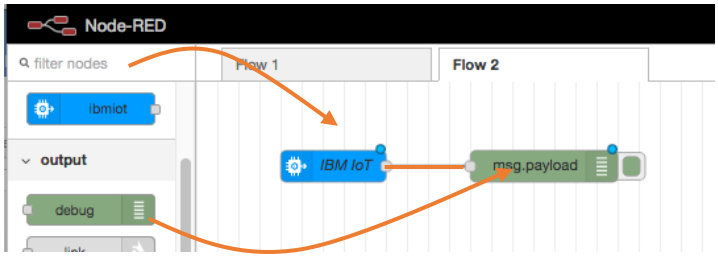
Note: Instead of using your desktop browser, you can use your smartphone.

__ 3. Identify your virtual device ID (top right corner) : copy it. You will use it in next section.
Warning: if you reload this page, the device ID changes.

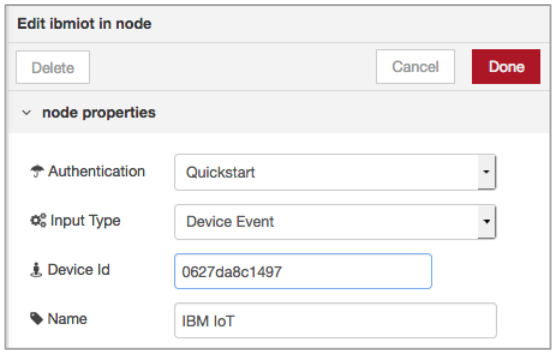
Section 2. Node-RED flow: creation & import

__ 1. Go back to Node-RED window

__ 2. From left panel, drag and drop nodes to the workspace
- Chose the Input node **ibmiot**
- Add an output **debug** node
- Link them

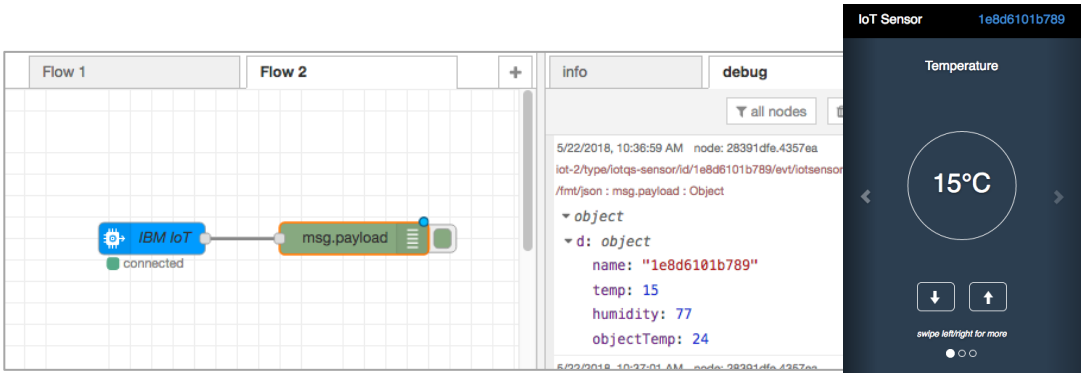


__ 3. Configure IBM IoT by double clicking on it :
- Authentication: Quickstart (means it is a simple authentication – for demo purposes)
- Device ID : <The value from Section 1, step 3 - Generated by the Simulator>

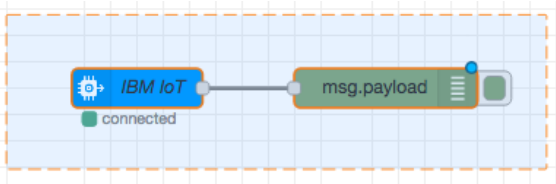


__ 4. Click **Done** & deploy your flow by clicking the **Deploy** button (top right).

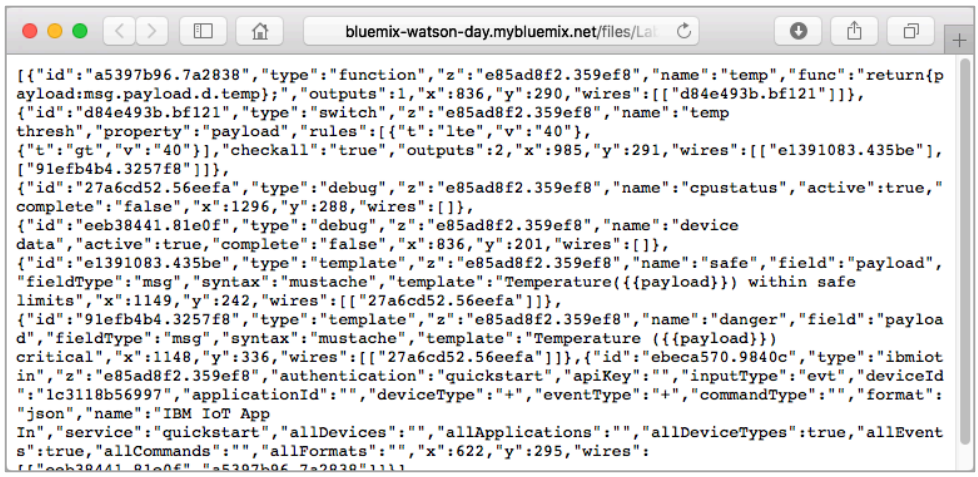
__ 5. Check the **Debug** Panel on the right side while you are playing with the sensor simulator. You should receive Device (sensor = web app. you opened in other window) data as the IBM IoT Node subscribed to this particular Device topic.



__ 6. Delete the whole flow by selecting all the nodes & pressing the 'Delete' key.

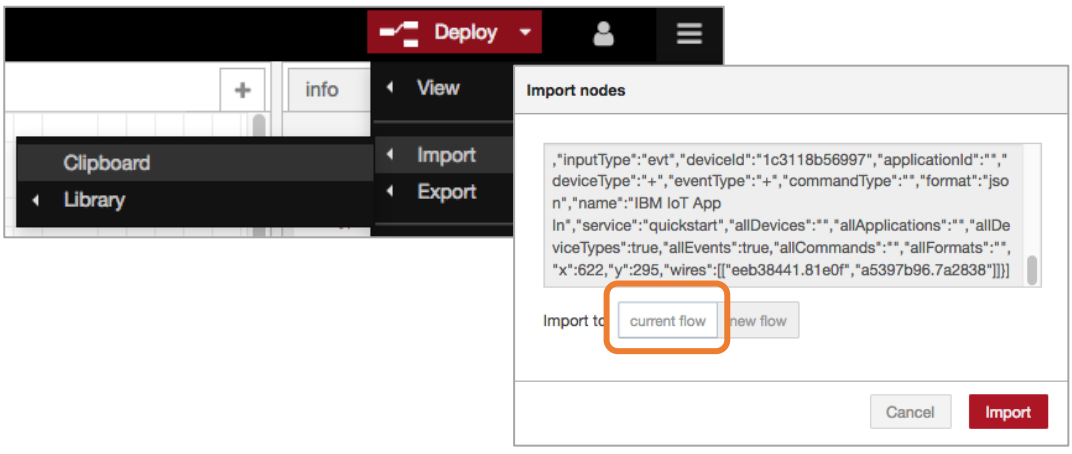


__ 7. Now import a new flow. A flow can be exported and imported using JSON file.
Open link http://ibmcloud-watson-day.mybluemix.net/files/Lab3_IoT_Flow.v2.1.json to display code in JSON format. You can also open the file you downloaded previously.



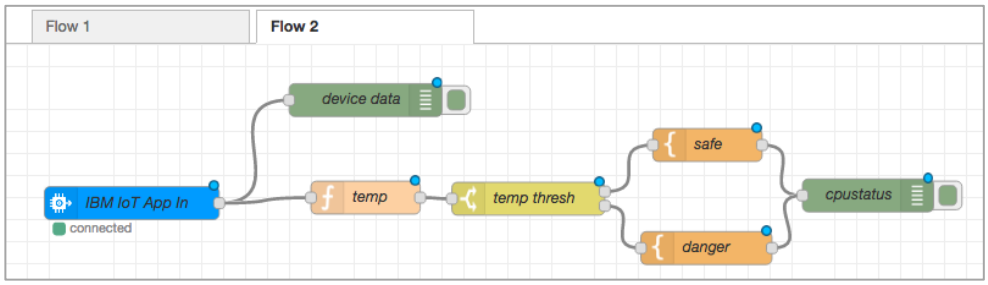
__ 8. Select all and copy it

__ 9. Click on the top right button near **Deploy**.
Select **Import**, **Clipboard** & copy/paste the content of the JSON file in **Current flow**.

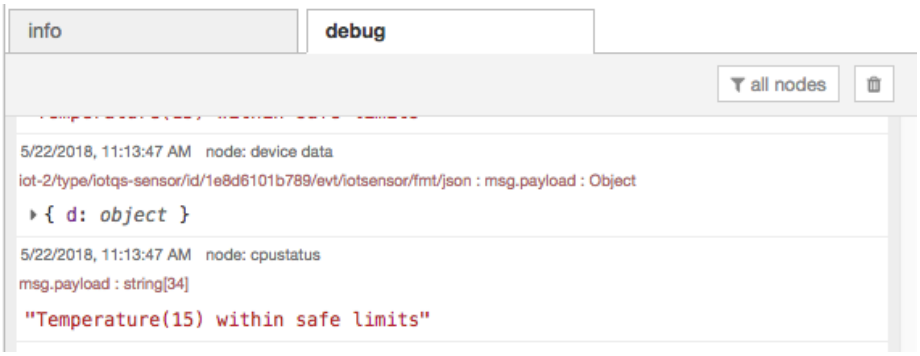


__ 10. Click on workspace to paste imported nodes

__ 11. Fill in the **Device ID** field in the **IBM IoT App In** node.



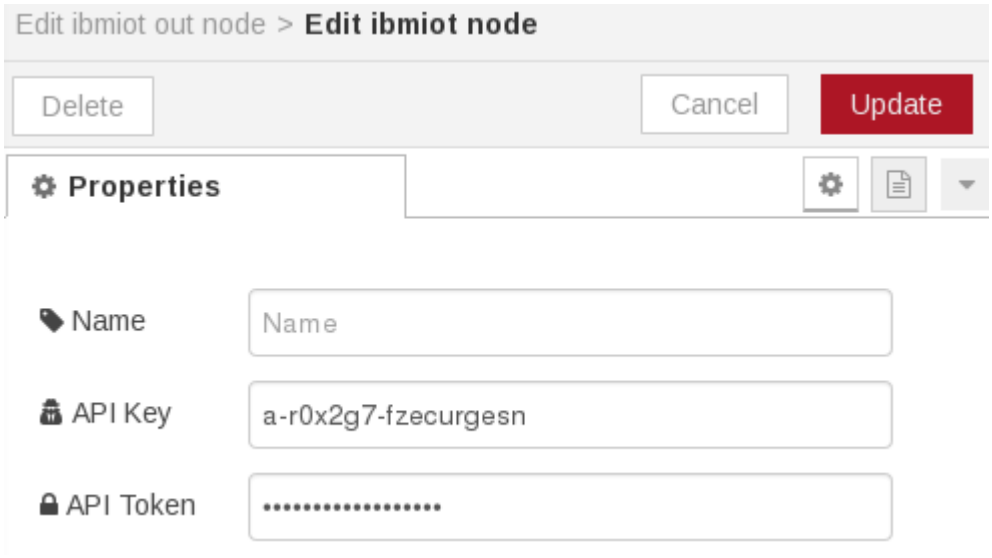
__ 12. Click **Deploy** to deploy the new Flow.
Modify the Device Temperature & check the **Debug** logs.

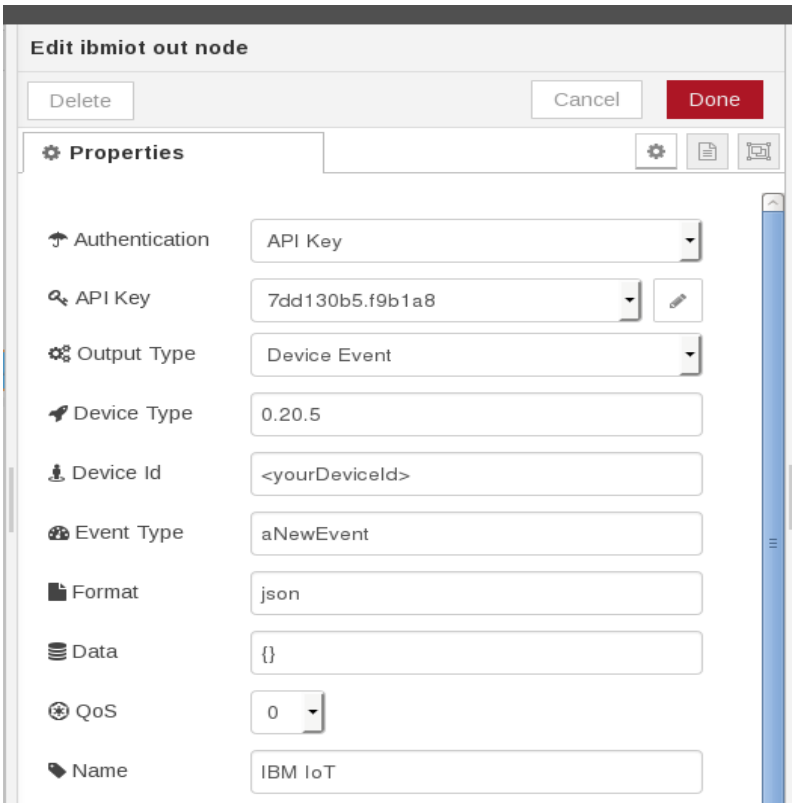


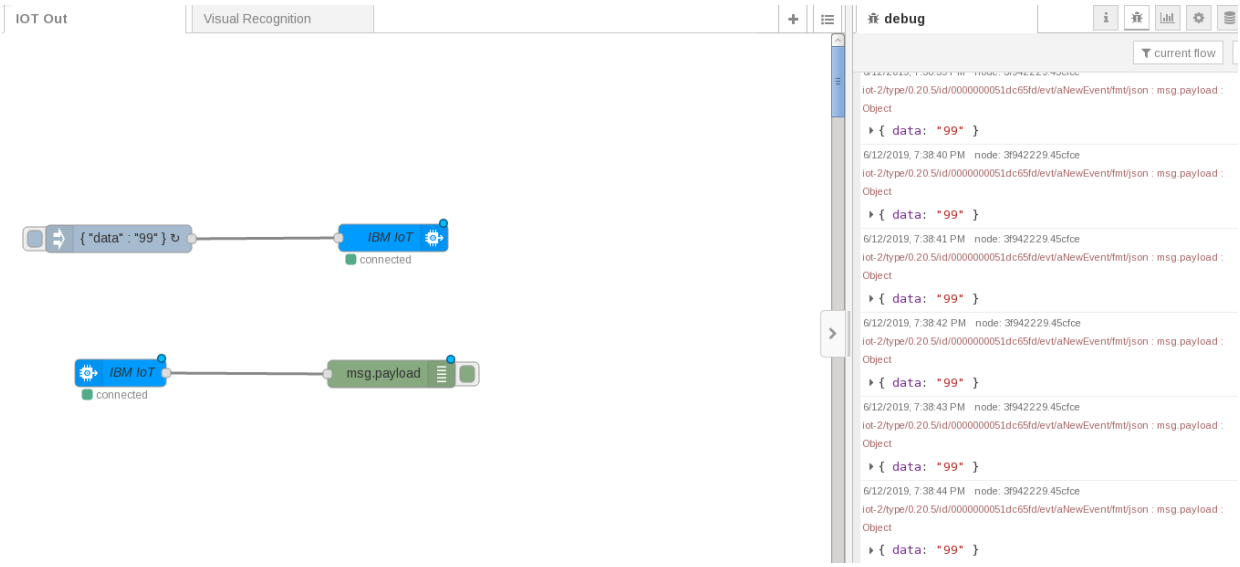
2. Node-RED on PI: Your first MQTT Simulator

MQTT is a pub/sub protocol: you are going to write your own device that will publish messages to the IoT Platform service (MQTT Broker).

Using the Credentials, given in the Labs menu of the web application at http://Raspberry314-POT.eu-de.mybluemix.net, create on the Raspberry PI directly a NodeRed flow simulating a device sending data at regular intervals.







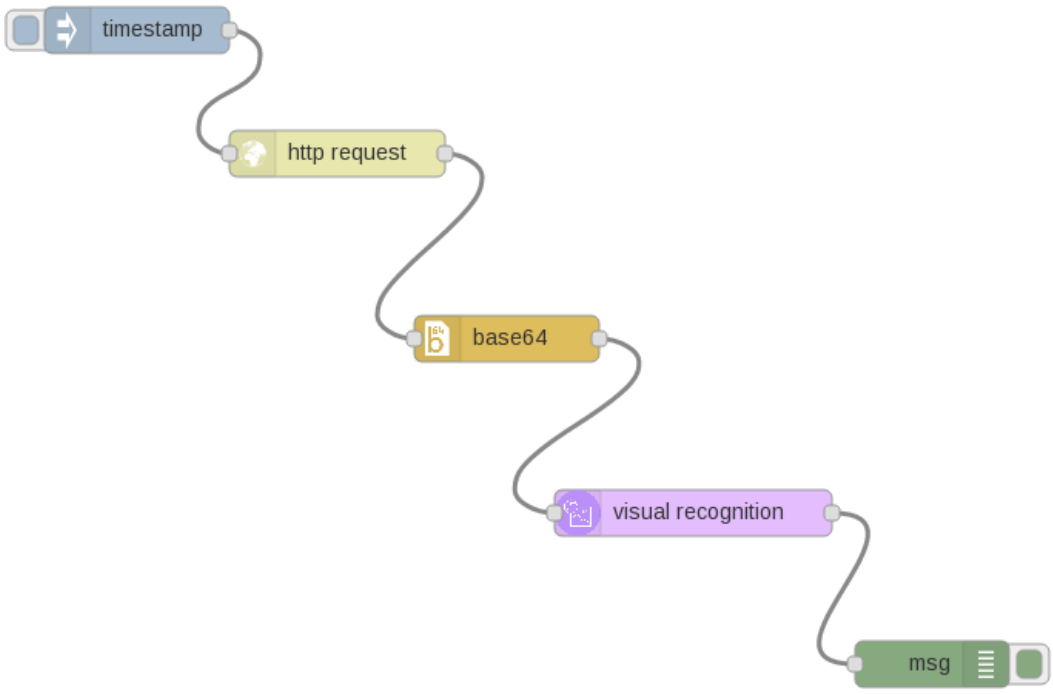
3. Cloud & Raspberry PI: Visual Recognition

Using the URL given in the Labs menu of the web application at http://Raspberry314-POT.eu-de.mybluemix.net, which returns the last image of one of the Raspberry Pis, analyze the picture with Watson Visual Recognition.

You first need to create a Visual Recognition service in your IBM Cloud account and bind this service to your existing NodeRed application.

Then create this flow below with the HttpRequest node using the URL provided to get the last image taken by one of the Raspberry Pis.

This flow requires the installation of the package base64 (Menu Manage Palette of NodeRed).

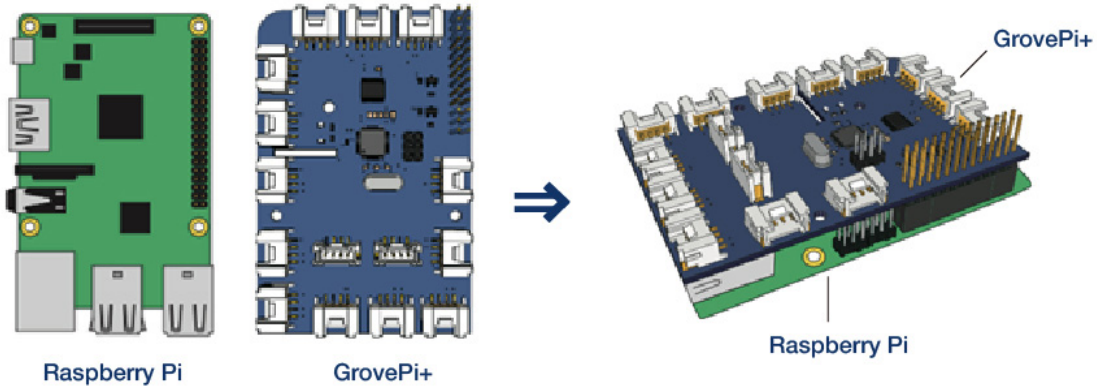


Export and import the flow on your Node-RED running on your Raspberry. Make sure all the nodes are properly installed on the target Node-RED. Adapt your flow (API Key, etc.) and test your flow.

4. Node-RED on PI: Grovepi Hat & Sensors (Optional #1)

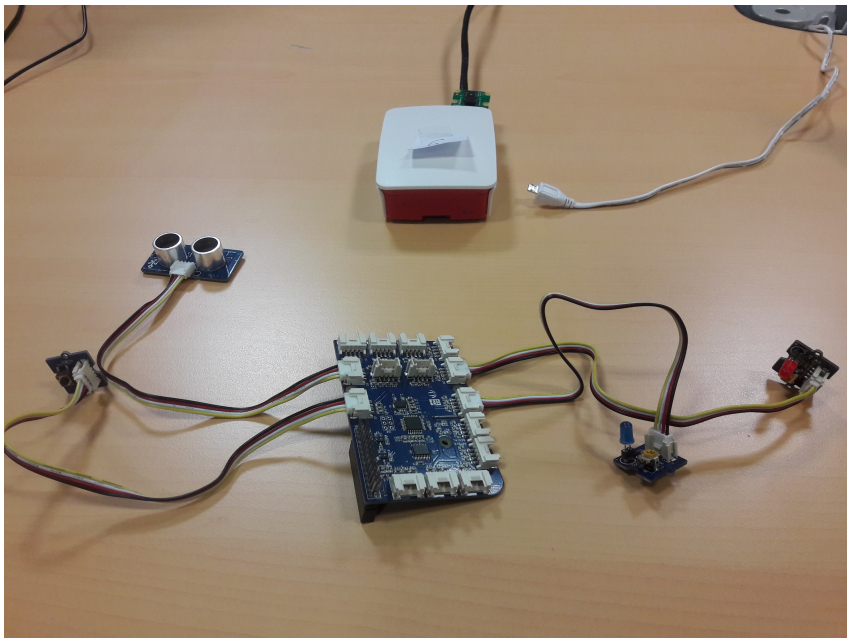
Section 1. GrovePi Hat Setup

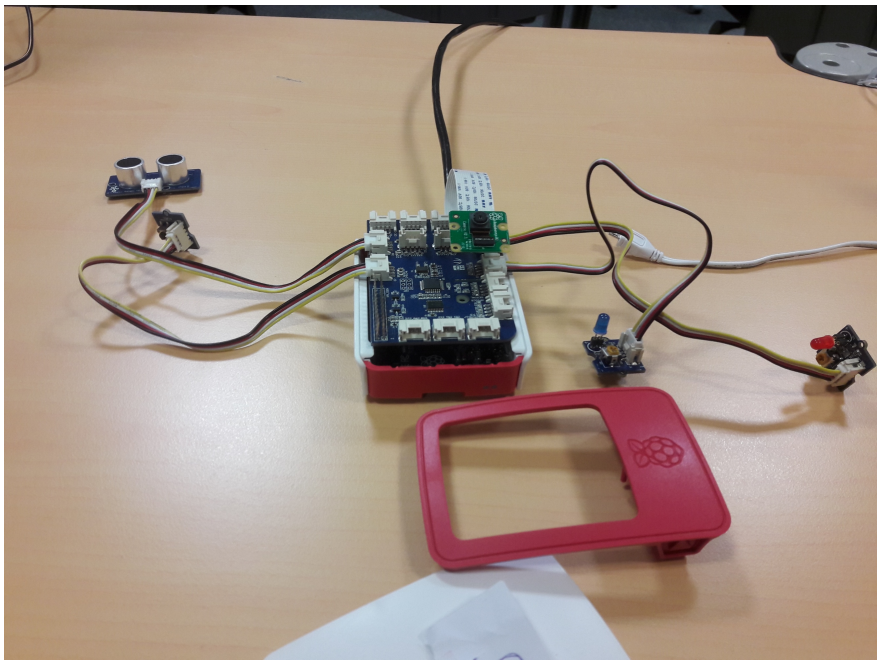
__ 1. Shutdown your Raspberry PI and connect the Hat as shown below



Source: https://www.dexterindustries.com/GrovePi/get-started-with-the-grovepi/

__ 2. Connect a few sensors on the Hat. Note that some ports are numbered as AXX (as Analogic) and others are numbered DXX as Digital.
LED, button, Ultrasonic sensors etc. are all Digital sensors.





__ 3. Reconnect your Raspberry PI : your Node-RED server should be back & running after a few seconds. Connect to Node-RED using http://<RASPI-IP:1880

__ 4. If not already installed, Install the Grovepi Nodes:
- In the top right Menu, select “Manage Palette” and browse the already installed nodes. If not in the list, go to the **Install** panel type “grovepi” in the search bar, and search for “node-red-contrib-grovepi”.
- Click on **install**

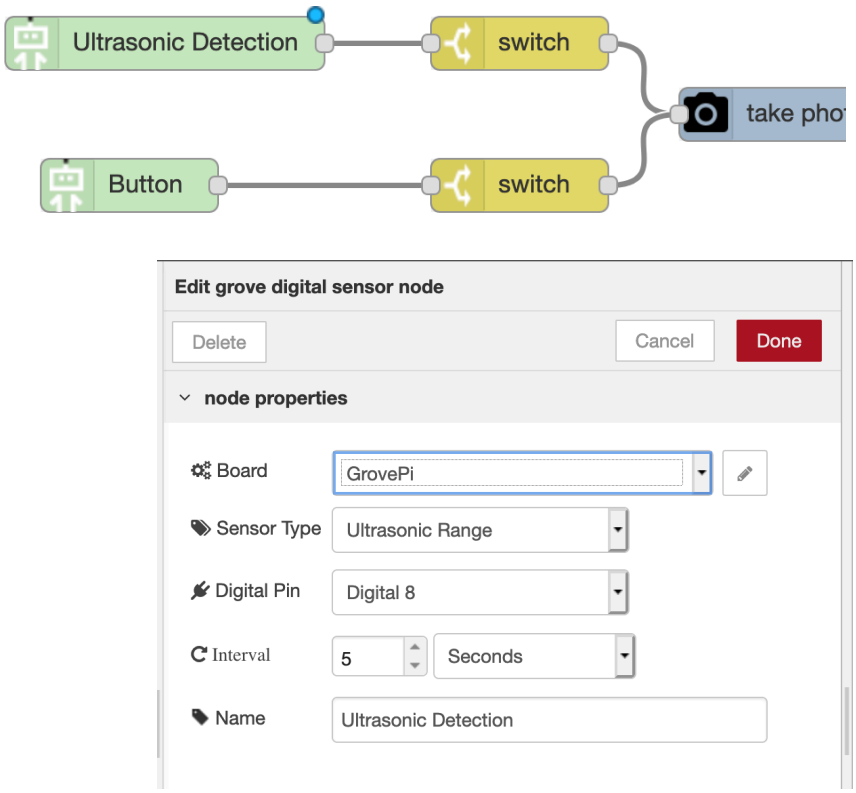
__ 5. If not done, install the Camera PI Nodes.

__ 6. Adapt your previous flow to create a Visual Recognition flow , using the Camera PI node instead of the previously used http get node.

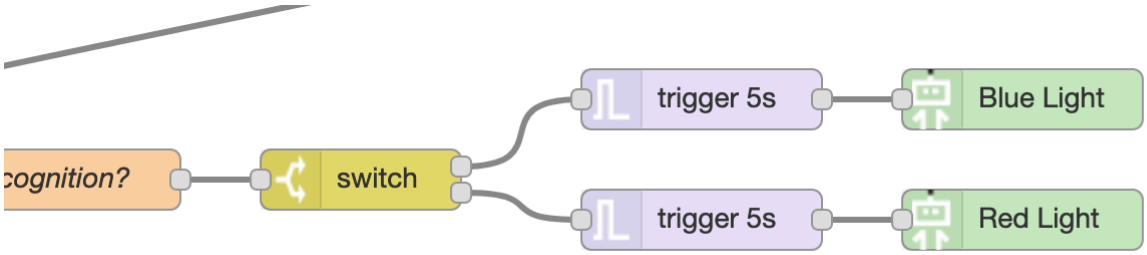
Section 2. First flow with Grovepi

__ 1. Adapt your previous flow, and replace the **Inject** node which triggers the execution of the flow by a set of Grovepi nodes. For example, trigger the photo capture using an ultrasonic sensor (presence sensing) and/or a simple pushbutton switch.

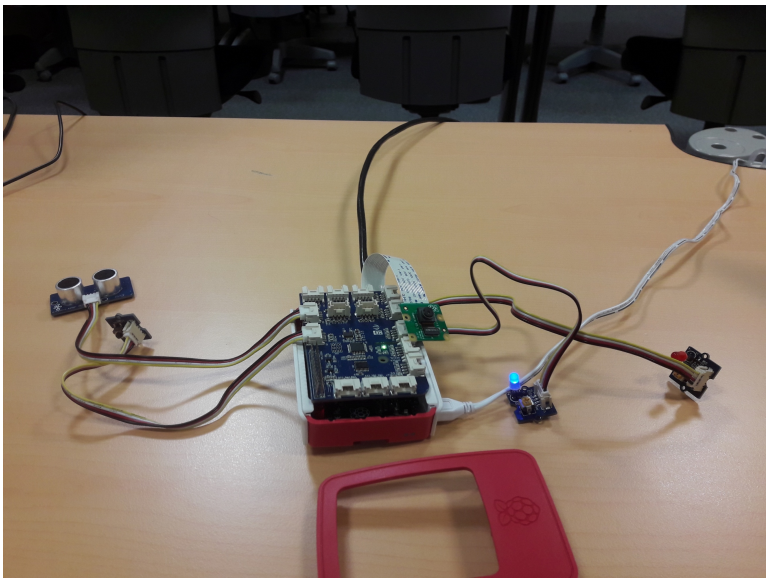
NB: An example of flow is available here on the **Lab Github** and can be imported using the top right menu, **Import** (from clipboard). Integrate this example with your existing flow.



__ 2. Use an output sensor like a LED for customizing an action based on conditions.



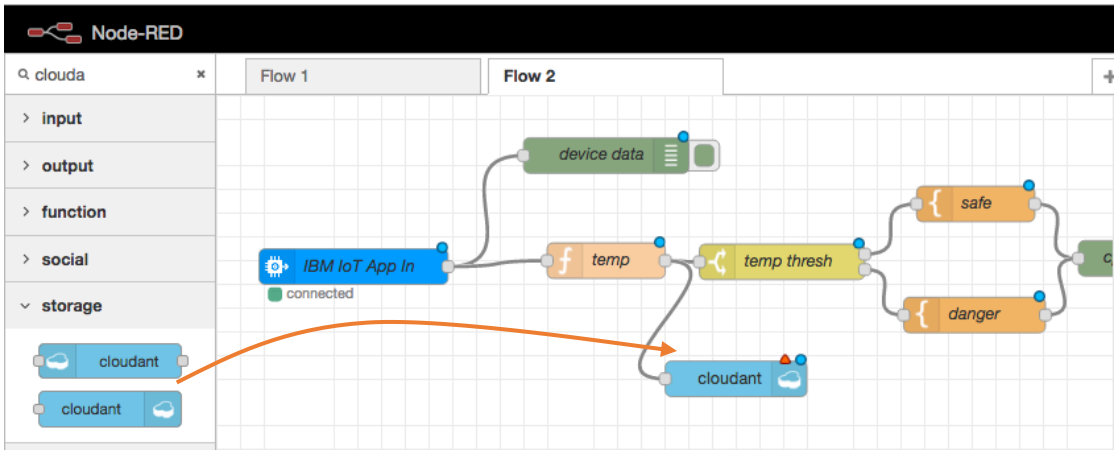
__ 3. Prototype testing & result sharing with other teams
Example: Presence detection, Visual Recognition with Watson, actions based on detections.



5. Insert IoT Data in Cloudant DB (Optional #2)

Let's insert the event data coming from the Device sensors in a Cloudant database!
Remember that you already have a Cloudant service deployed for Node-RED. You will use it to store your data.

__ 1. Add a Cloudant Node (**Cloudant OUT** node in the **Storage** Category) & link it to the **temp** function node

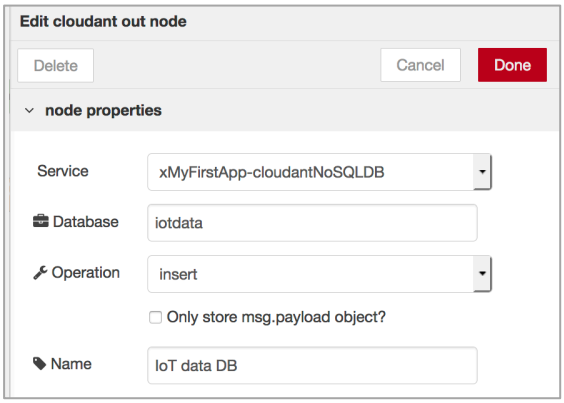


__ 2. Configure Cloudant client in Node-RED.
2 Options:
A) Node-RED On IBM Cloud
B) Node-RED on any "remote" device (Raspberry PI, server…)

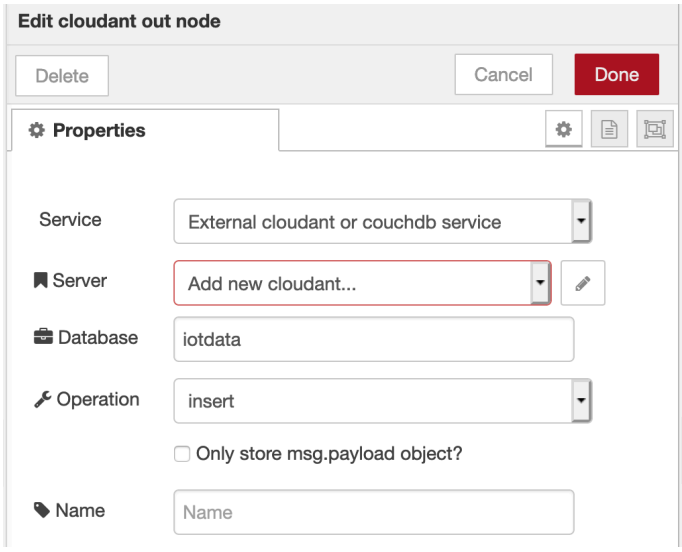
A) If running on IBM Cloud (Cloudant service auto-detection)

- Service : Cloudant service name bound to your Node.js runtime.
  As Node.js is already bound to a Cloudant Service, the service name should appear in the Drop-down list.
- Database: name of your choice (lower case)
- Name (node): name of your choice

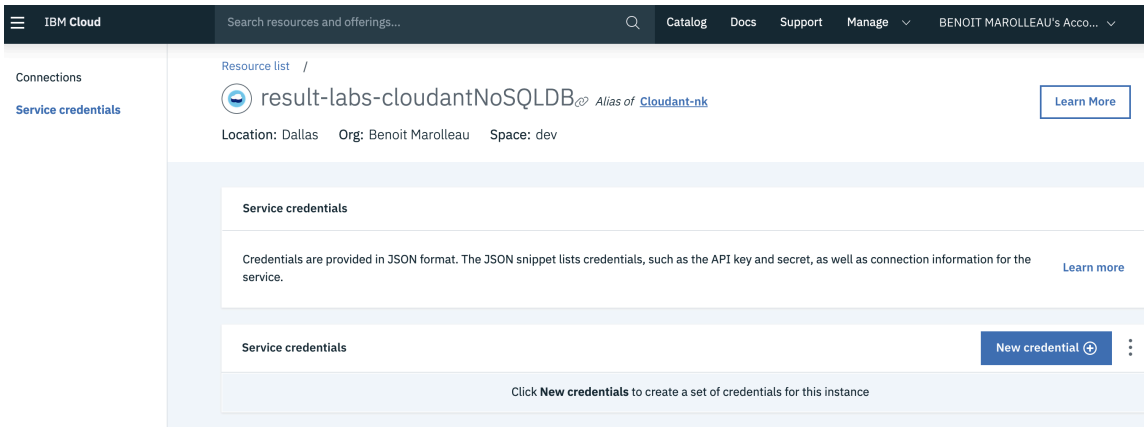
Click **Done**.



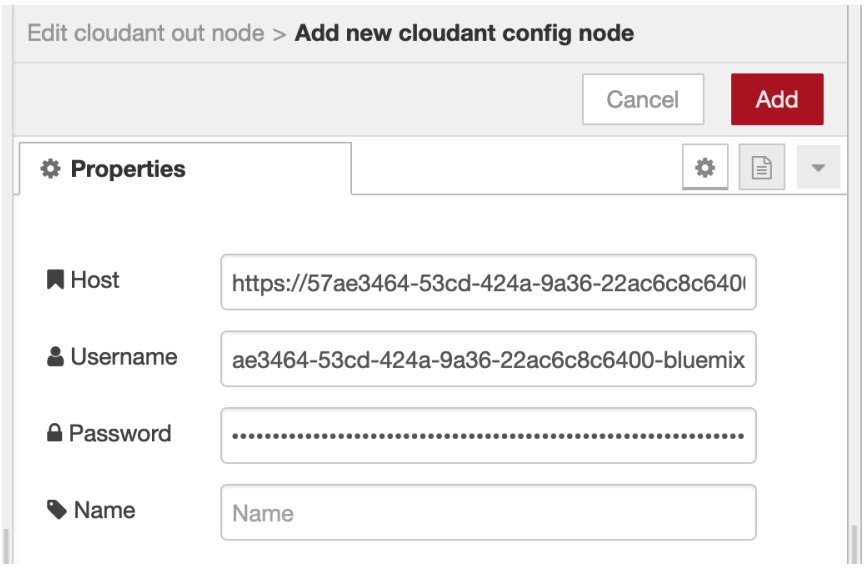
B) If running on a remote device (Raspberry PI …)



- In **Service**, select “External cloudant or couchdb service”
- In **Server,** click on the button on the right
- On IBM Cloud, Go to the Resource list page , then on the Cloudant database service running in your IBM Cloud workspace.This Cloudant Database service was created in your IBM Cloud account, in the first exercise.
- On the service page, Go to **Service Credentials .**
  If there is no credentials available, click the **New Credentials** button, click **Add**



- Click on **View Credentials** and copy/paste the information to your Node-RED config node as shown below

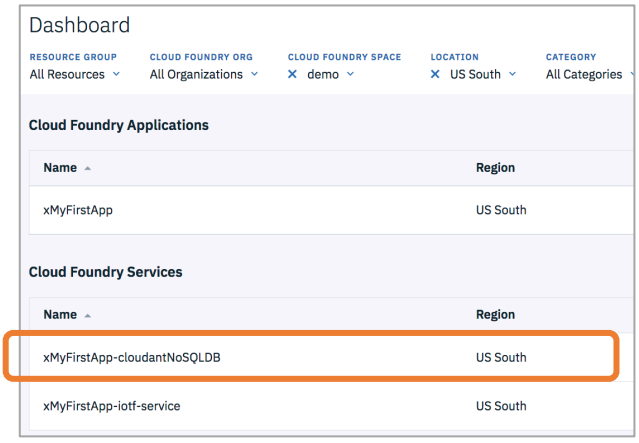


Host = url field in the Credentials.
Username and **Password** = username, password in the Credentials.
Click **Add**

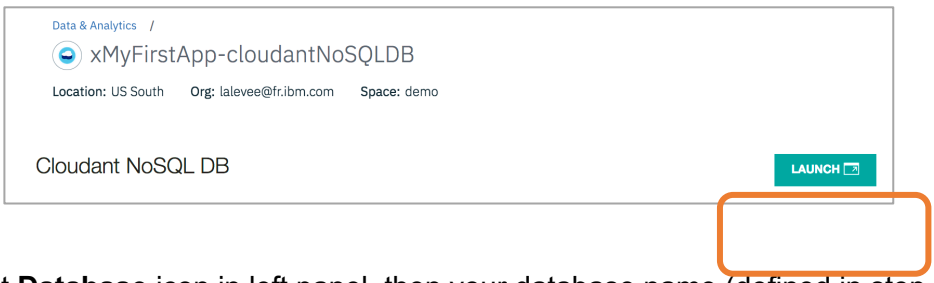
- Database: name of your choice (lower case)
- Name (node): name of your choice.
- Click **Done**

__ 3. Deploy your new flow (**Deploy** button)

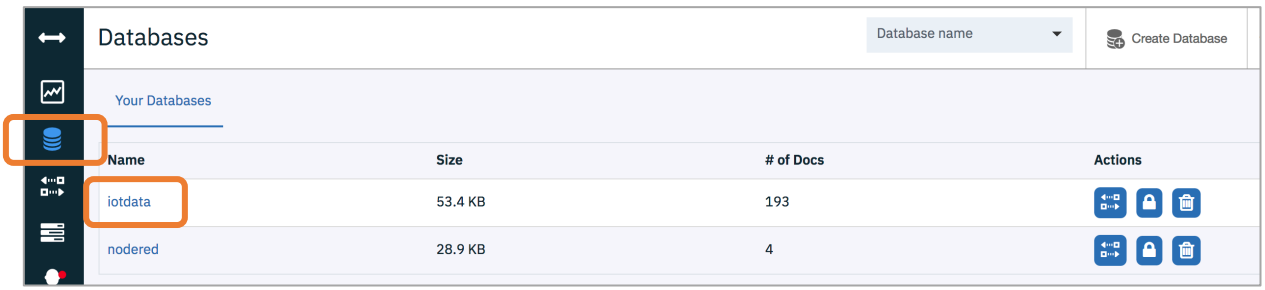
__ 4. From your IBM Cloud Dashboard (IBM Cloud window of your browser), start the Cloudant dashboard by clicking on the line of Cloudant service



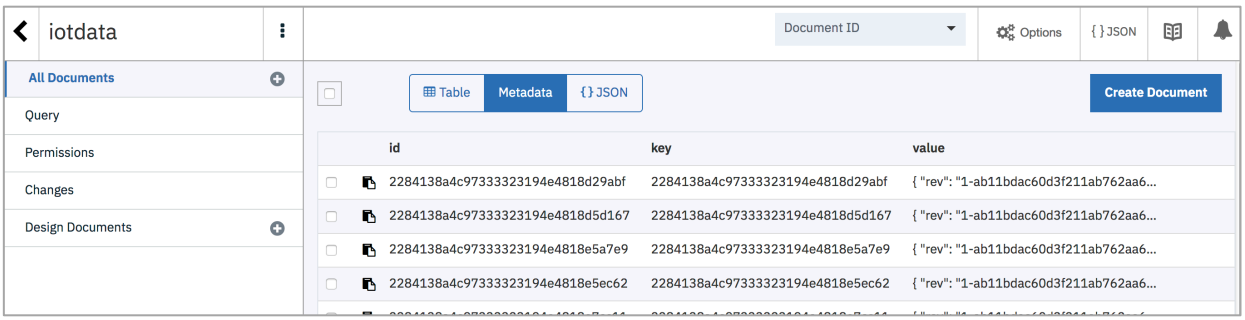
__ 5. Click **Launch** button to start Cloudant console



__ 6. Select **Database** icon in left panel, then your database name (defined in step 2).



__ 7. Have a look to the inserted data in the database. Click on record to see content.



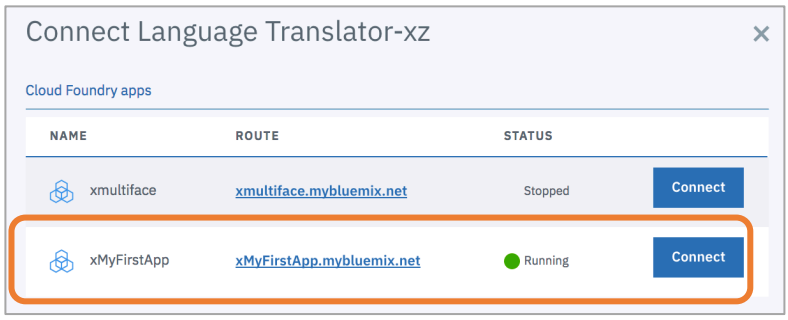
6. Process IoT Data with Watson Language Translator (Optional #3)

__ 1. In IBM Cloud windows, add a « Watson Language Translator » service to your existing Node-RED application. From **Catalog**, click on **Language Translator** in Watson category.
Select **Lite** plan, then click **Create**. Service is deployed.

If you use Node-RED outside IBM Cloud, please skip steps 2, 3, 4, 5, 6 as they are related to service binding (Node-Red App ←→ Watson Service)

__ 2. From Language Translator dashboard, **Connections** menu, click on **Create connection**.

__ 3. Connect service to your Node-RED application

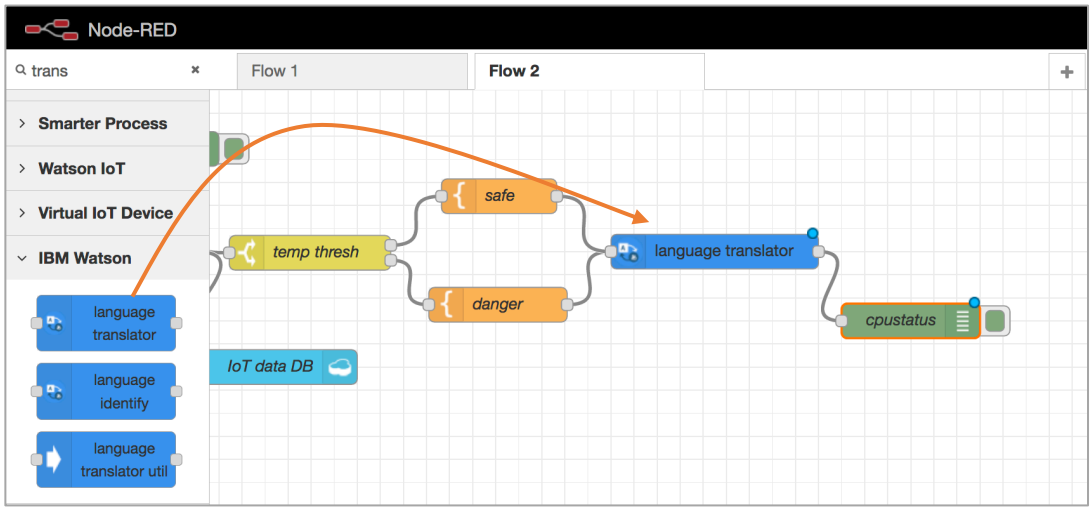


__ 4. Accept the restage step to actually bind the service to the app.

__ 5. While it is restaging (~3 minutes), take a look at **Service Credentials**. This information is useful if you want to invoke your Watson Service from any program (running in IBM Cloud or outside IBM Cloud, on a Raspberry PI for example).

__ 6. Go back to Node-RED window.
If you get a connection error message, your application restaging is not finished. Wait. Try to refresh page.

__ 7. Go back to the Node-RED environment.
Add **Language translator** node and link it between the template (*safe & danger)* & debug nodes (*cpustatus*).



__ 8. Configure the Watson language translator node (2 options: A or B)

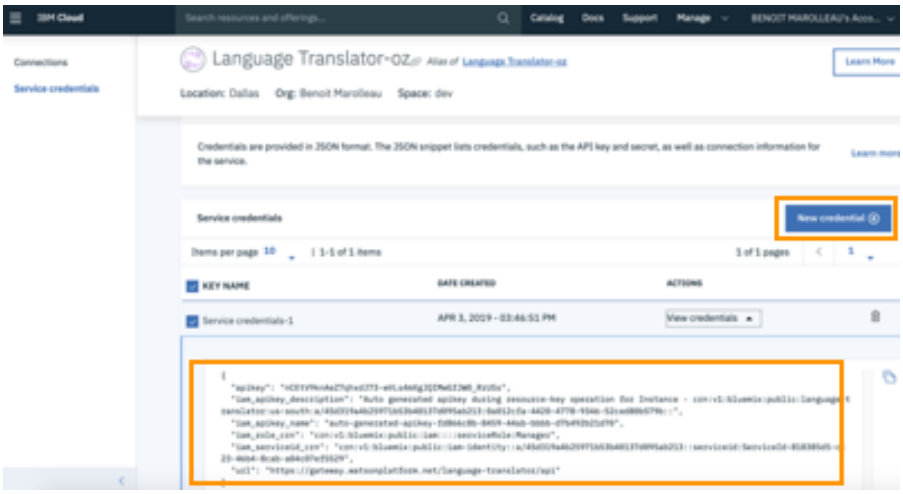
A) If running Node-RED on IBM Cloud,

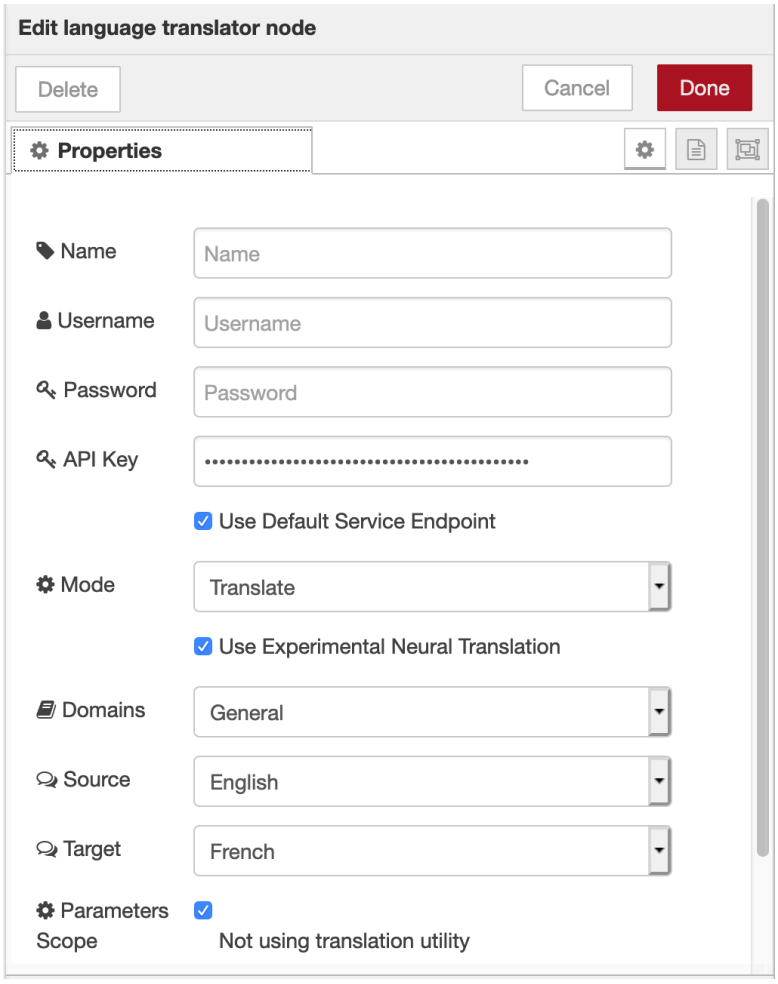
Your Watson Service is connected/bound to your Node-RED App, you don't have to set any credentials on the Watson Node.
Note: The user/password fields are not necessary & do not appear in the node settings if a Watson Language Translator service is properly bound to Node-RED application.

- Name (of your choice)
- Mode: **Translate**
- Domains: **Conversational**
- Source: **English**
- Target: **French** (or Spanish, Portuguese & Arabic)

B) If Node-RED running on a remote location (server, Raspberry…)

Go to your IBM Cloud dashboard, click on your Watson language translator service, **Credentials.** Copy the API Key





- Name (of your choice)
- Mode: **Translate**
- Domains: **Conversational**
- Source: **English**
- Target: **French** (or Spanish, Portuguese & Arabic)

__ 9. **Deploy** your flow & check the logs (**debug** tab).

7. Create a dashboard application in Node-RED (optional #4)

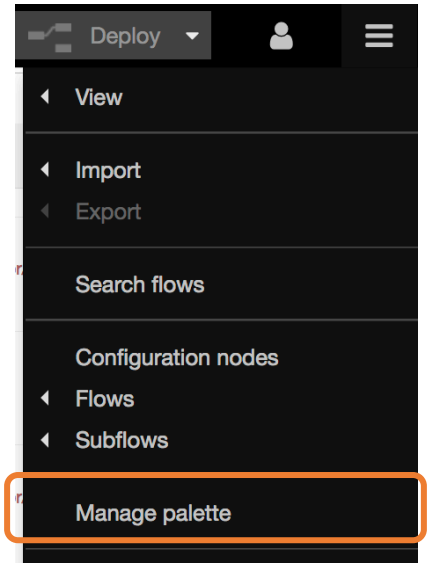
Section 1. Import Node-RED Dashboarding capability

The required file used during this lab can be downloaded the JSON files from

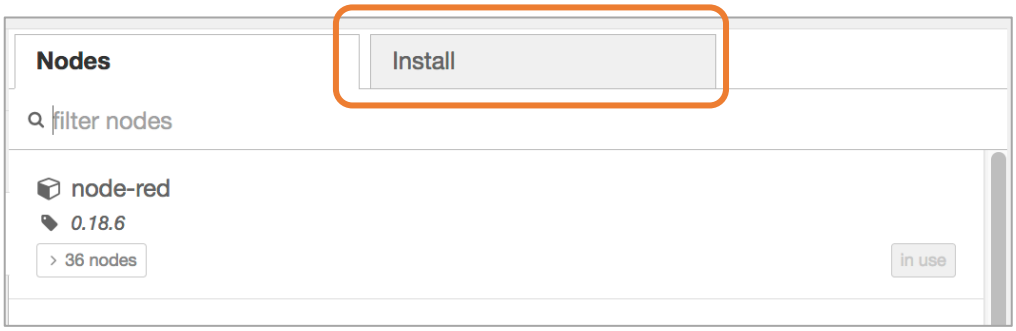
- http://ibmcloud-watson-day.mybluemix.net/files/Lab3_IoT_Flow.v2.1.json
- http://ibmcloud-watson-day.mybluemix.net/files/Lab3_IoT_Dashboard.v2.1.json

__ 1. At the top right-hand side of the page, click the 'burger' menu:

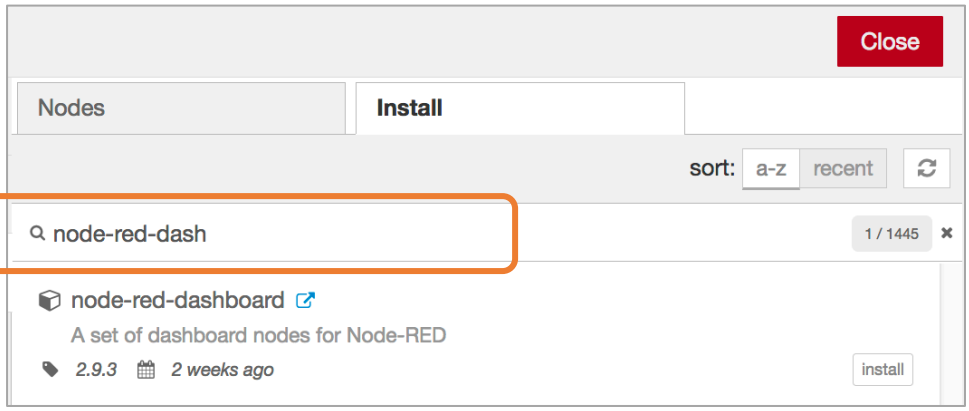
__ 2. Click **Manage palette**:



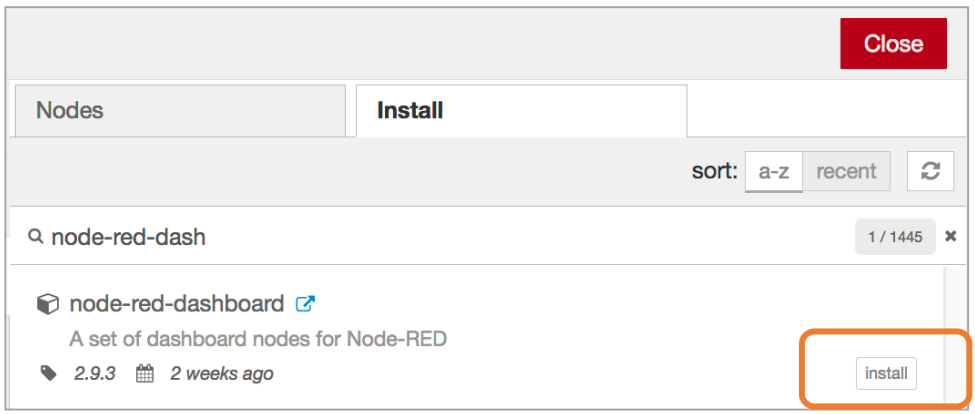
__ 3. In the sidebar that appears on the left-hand side of the page, click the Install tab:



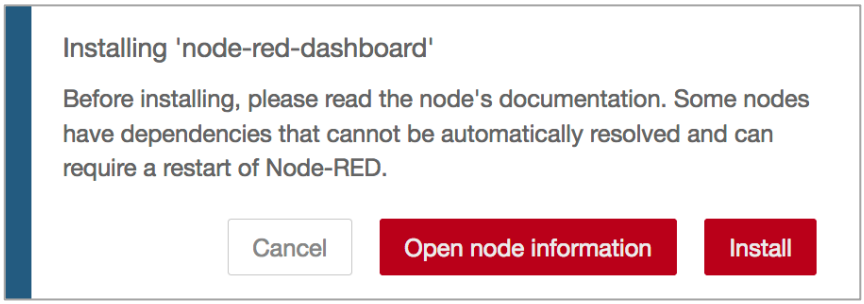
__ 4. In the search field, type **node-red-dash**:



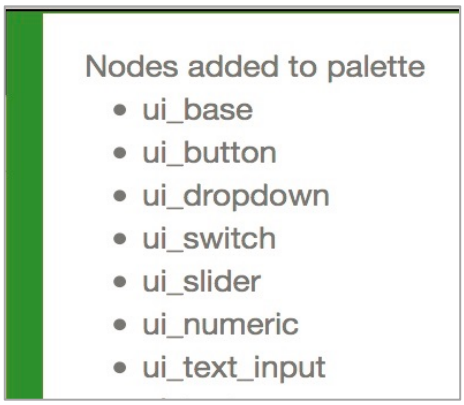
__ 5. Next to the nod-red-dashboard result, click **install**



__ 6. Click **Install**

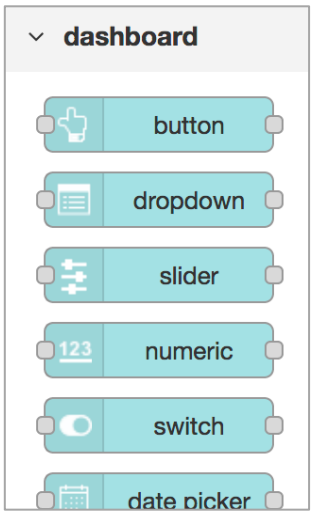


__ 7. Wait for the new nodes to be deployed.
A popup shows which nodes have been installed.

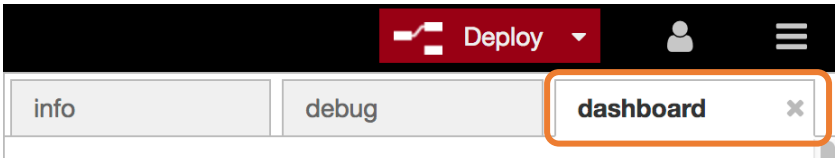


__ 8. Click **Close**.

__ 9. Note the additional dashboard nodes on the palette : **Dashboard** category.



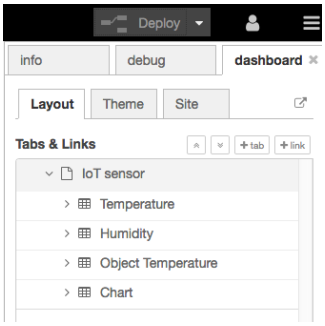
__ 10. Note also that there is a new dashboard tab in the right-hand sidebar:



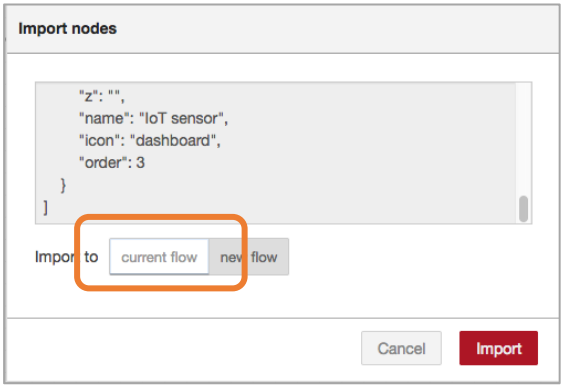
TIP: This dashboard tab may be used to add new tabs, menus etc. to the visualization dashboard. There are also two available themes – light and dark.

Section 2. Create a simple Node-RED Dashboard

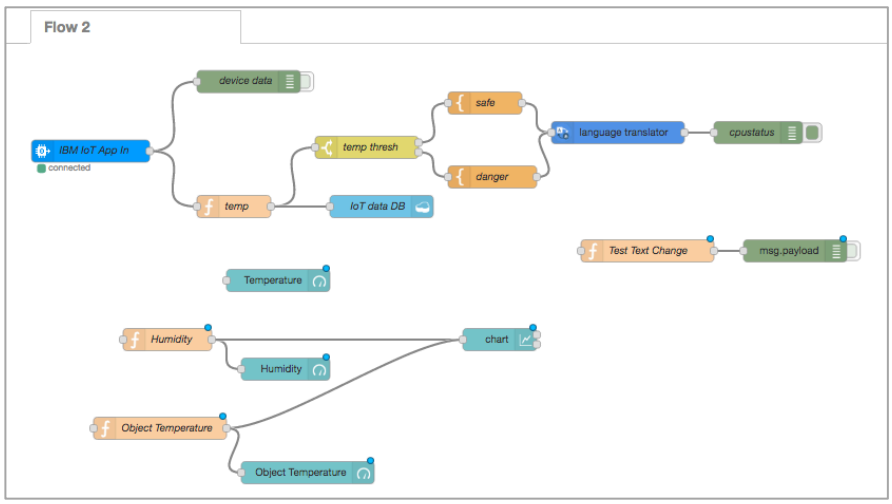
In this section, you will create a simple dashboard for sensor data using new dashboard nodes installed.



__ 1. In the current Node-RED tab, import the file named **Lab3_IoT_Dashboard.v2.0.txt** (previously downloaded): **Menu** > **Import** > **Clipboard**.
Click **current flow** then **Import**.

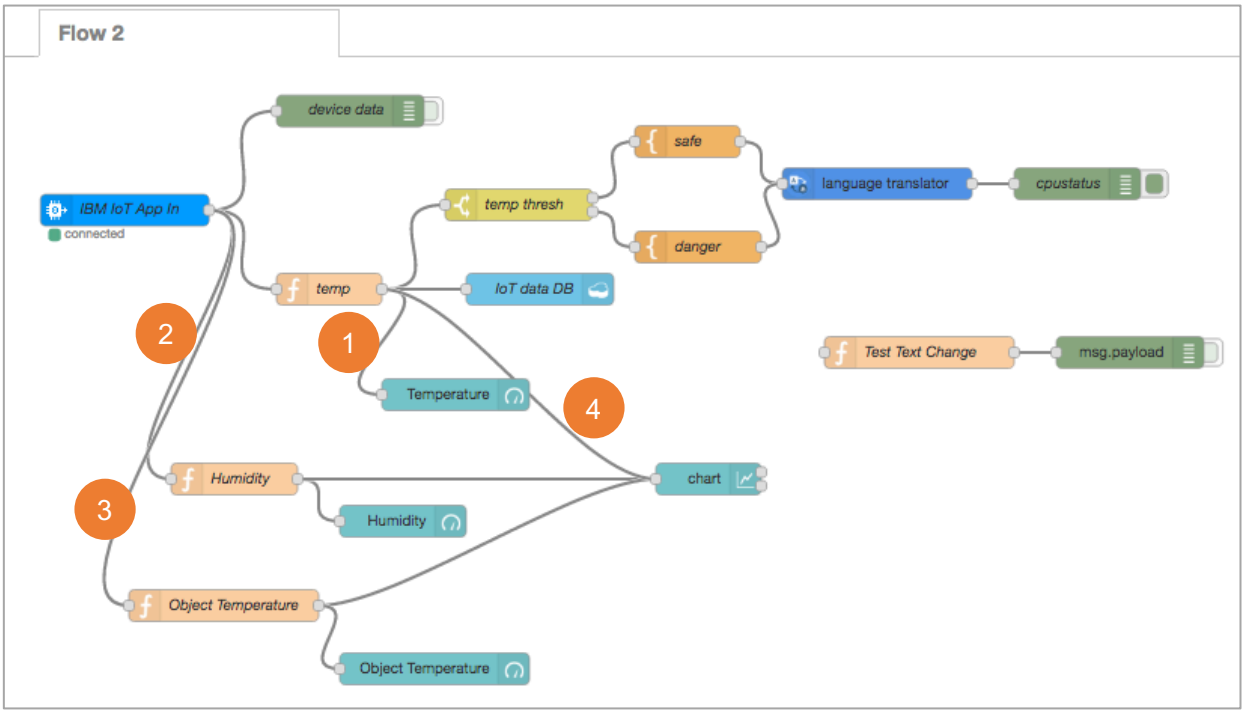


__ 2. Click on workspace to paste new nodes.



__ 3. Connect new nodes to existing…

1. **Temperature** to **temp**
2. **Humidity** to **IBM Iot App In**
3. **Object Temperature** to **IBM Iot App In**
4. **chart** to **temp**

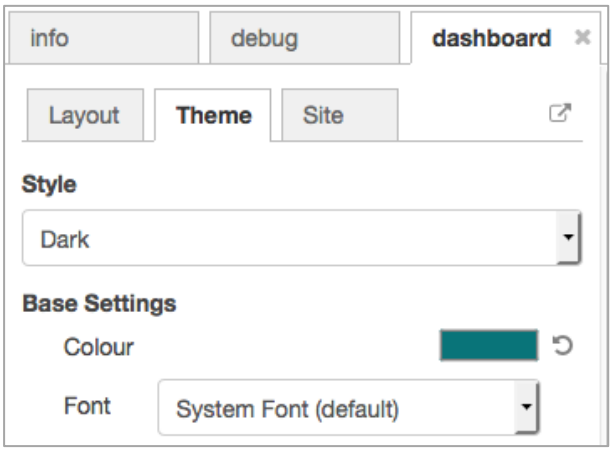


__ 4. Deploy the flow: **Deploy**

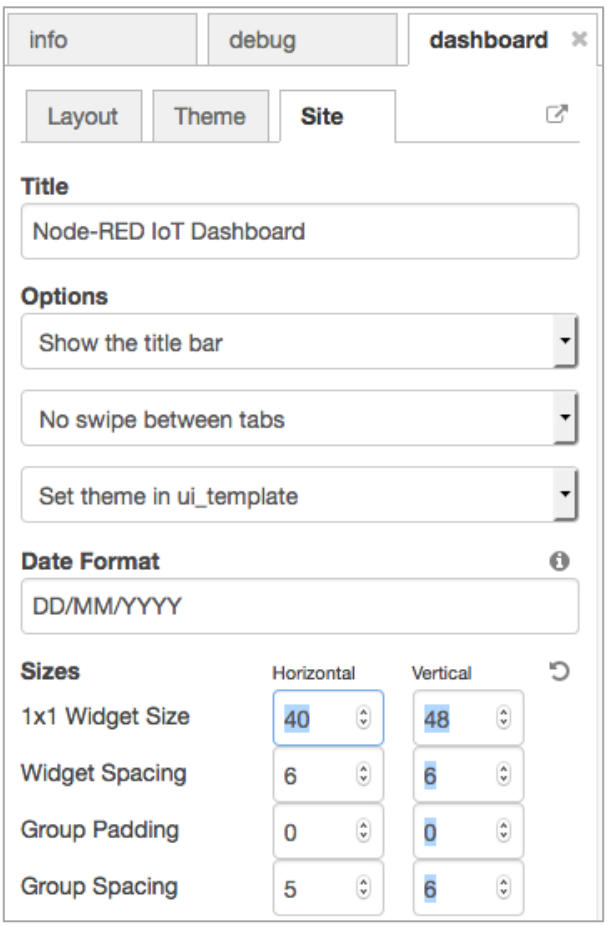
__ 5. Connect to **http://<YOUR_APP_HOSTNAME>/ui** (URL generated by Node-red-dashboard node) to see your new dashboard. Change value on virtual sensor app. to see impact on gauges and lines.

__ 6. Customize your dashboard using **Dashboard** tab

- **Theme**: use Dark style



- **Site**: as below, change **Title**, **1x1 Widget Size**, **Group spacing**



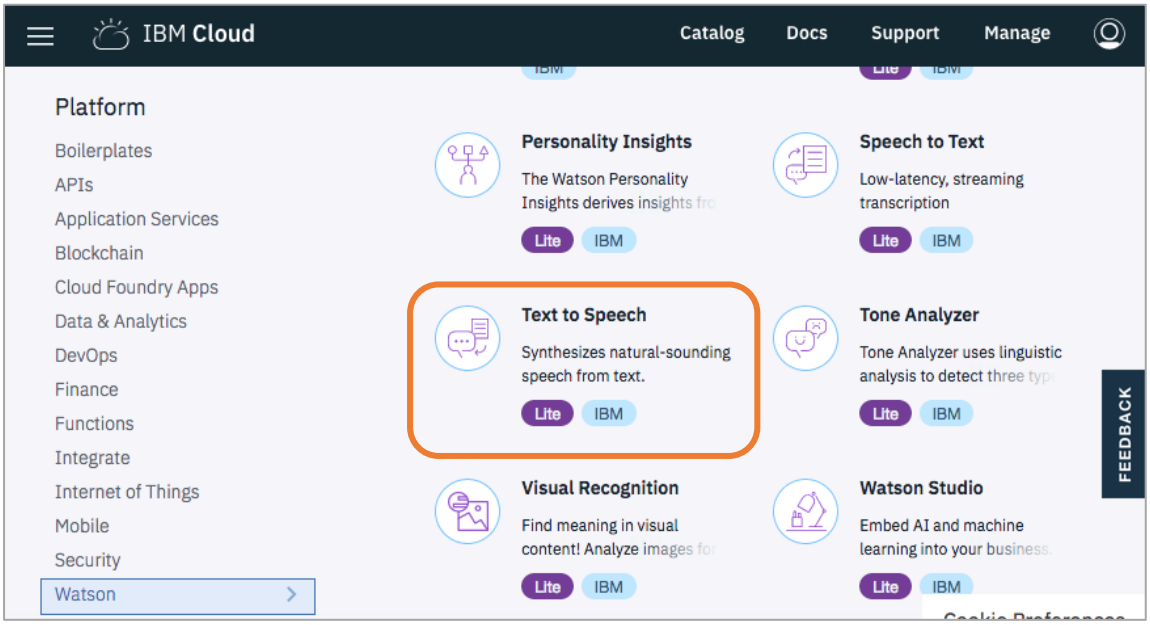
__ 7. Deploy and check your dashboard.

Section 3. Add voice alert on dashboard

In this section, you will add a voice node allowing your app to tell say message when temperature change. To do that, you will deploy a new Watson service: Text to Speech.

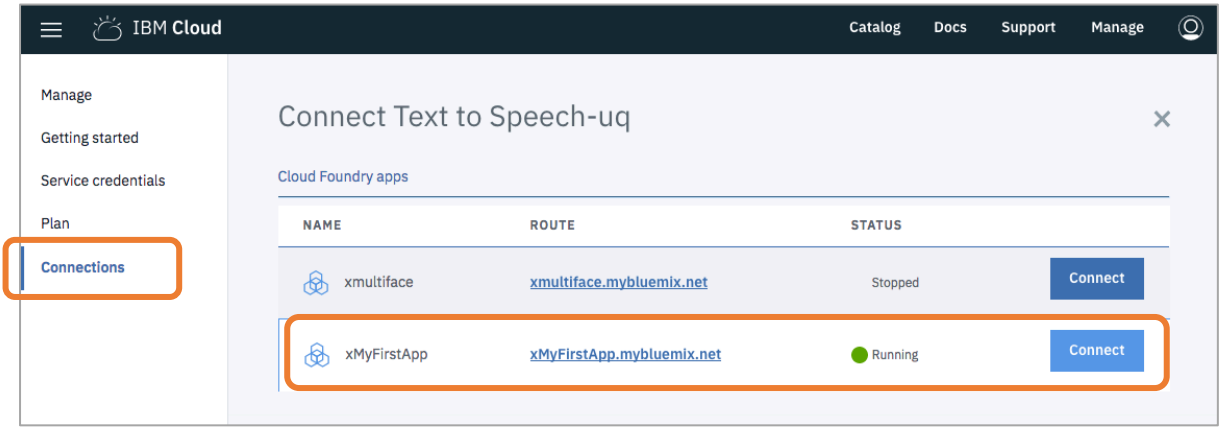
Note: If you don't run Node-RED on IBM Cloud, you will have to copy the (Watson) service credentials and use them in each (Watson) node configuration for accessing the service remotely. The instructions below are designed for running on IBM Cloud.

__ 1. In the IBM Cloud window, click **Catalog** then **Watson** category.
Click on **Text to Speech** service to create your own instance.

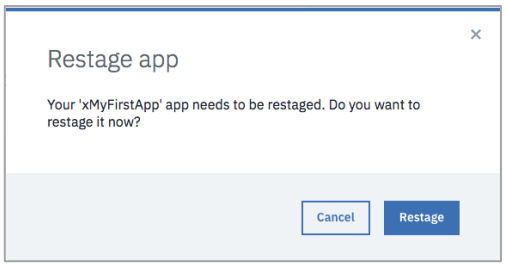


__ 2. Enter a name for your service and click **Create**.

__ 3. If running on Node-RED on IBM Cloud:
On new service dashboard, click **Connection**, and connect Text to speech service to your Node-RED application: **Connect**.

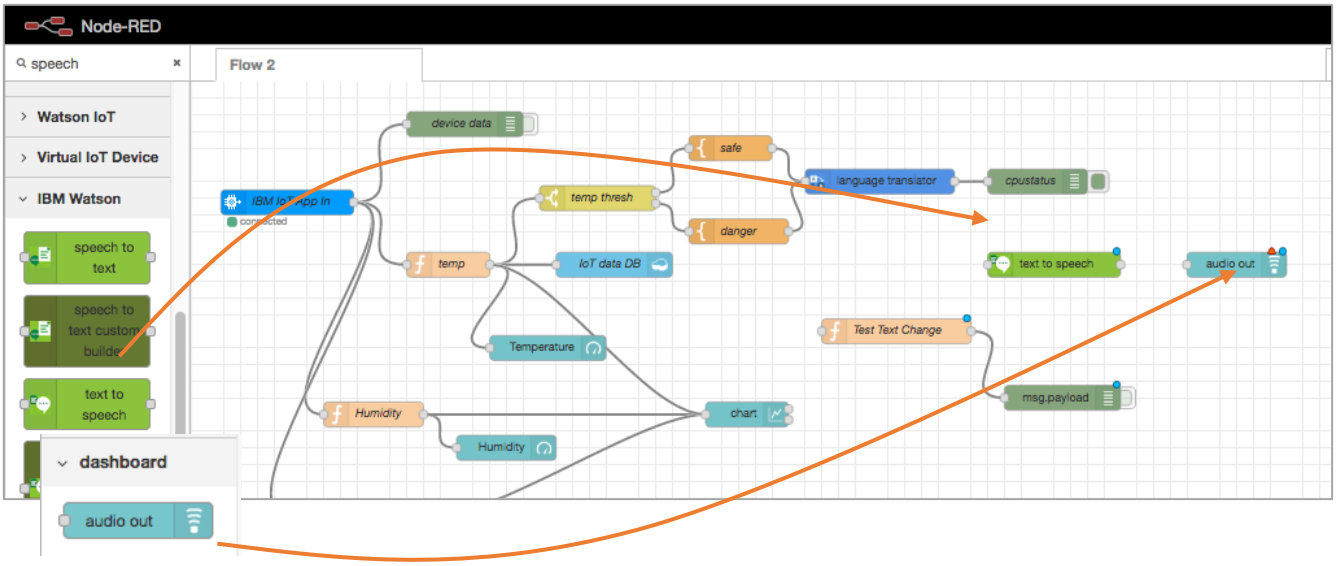


Accept to restage application and wait for your Node-RED application to restart.

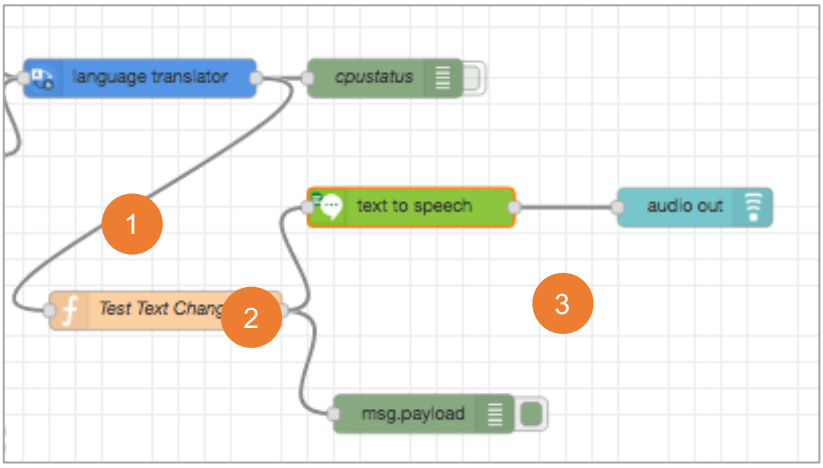


If running outside IBM Cloud, copy the Watson Service credentials (API key…) and use them in Step 8 – node Configuration. Step 4 is only for IBM Cloud.

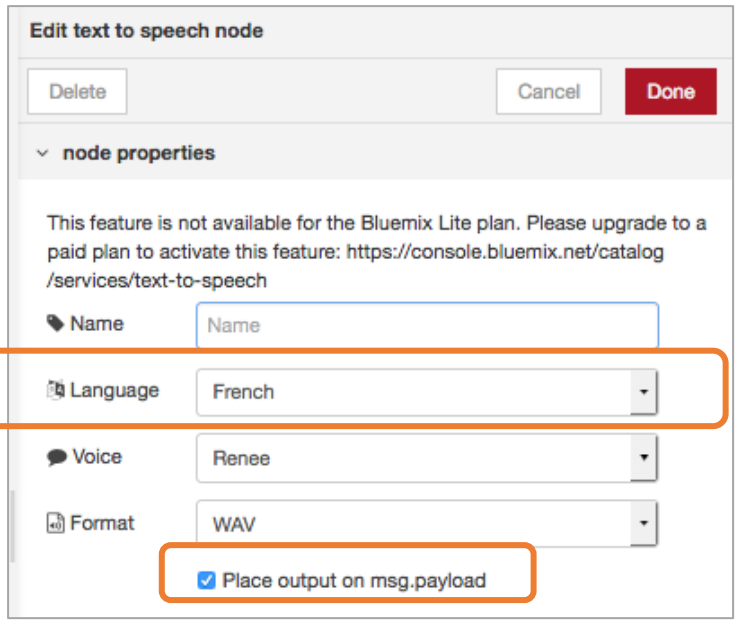
__ 4. When restarted, in Node-RED environment, add an **Audio out** node (category **Dashboard**) and a **text to speech** node (category **Watson**) to your flow



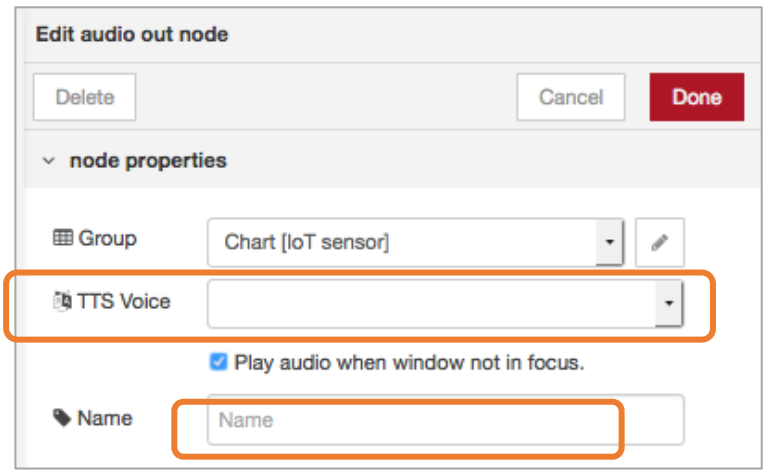
__ 8. Connect nodes as below
1. **Test text change** to **language translator**
2. **Test text change** to **text to speech**
3. **text to speech** to **audio out**



__ 9. Configure **text to speech** node to
- use the language your translated to (as configured before in **language translator** node)
- place the output on msg.payload (check box)



__ 10. Configure **audio out** node
- Set **group** to **chart [IoT sensor]**
- Check **Play audio when window not in focus**



__ 11. Deploy the flow: **Deploy**

__ 12. Can you hear something?
Try to change temperature in virtual sensor app.

_____ END OF LAB _____

8. Appendices: Architecture & Schema on IBM Cloud