IBM Cloud & Watson Labs
Lab 3

May 2018

Benoit Marolleau < benoit.marolleau@fr.ibm.com>
Architect – IBM Montpellier Client Center
IBM France

http://bluemix-watson-day.mybluemix.net

Labs – Before Starting

- Access to IBM Cloud (requires a valid IBM Cloud Account)

  https://console.ng.bluemix.net

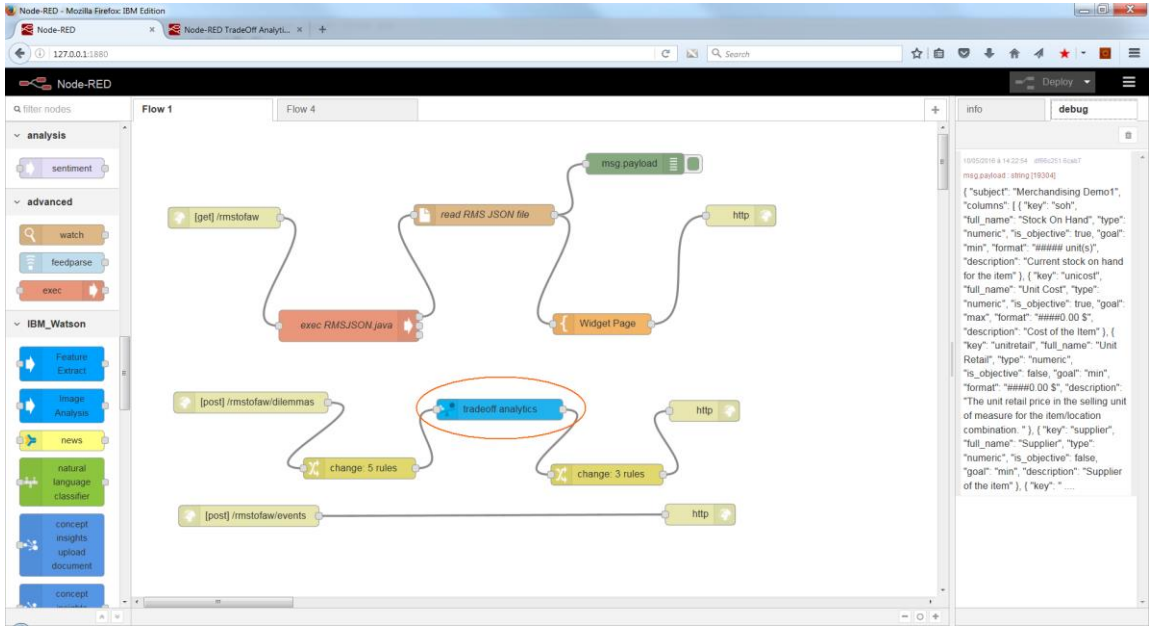
Ensure you have enough free resources (GB / #Services) in your IBM Cloud Organization / Spaces to run the lab exercises. *If you encounter a resource contention (Error Message saying you are out of resources), clean up your Spaces by deleting existing Apps or Services.*

- If you have a problem with an existing IBM Cloud Account (Corporate or student account etc.)

→ a workaround for this session is to create a new Account using another valid email address: https://console.bluemix.net/registration

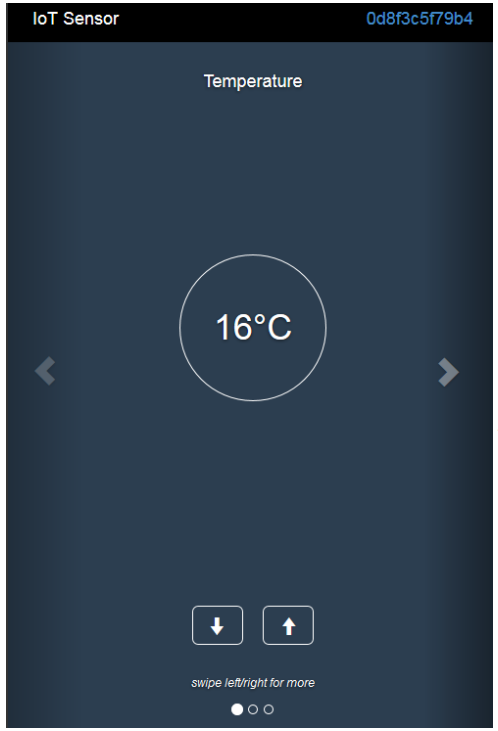
Lab 3 - Node-RED in a nutshell

- Node-RED is a software tool developed by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things.
- Browser-based flow editor with a wide range nodes in the palette.
- JavaScript functions with built-in library for re-use.
- Node-RED allows you to quickly build a restful web services based on REA
- Node-RED is now part of the JS Foundation.
- Node-RED is also an IBM Cloud service.
- https://nodered.org/

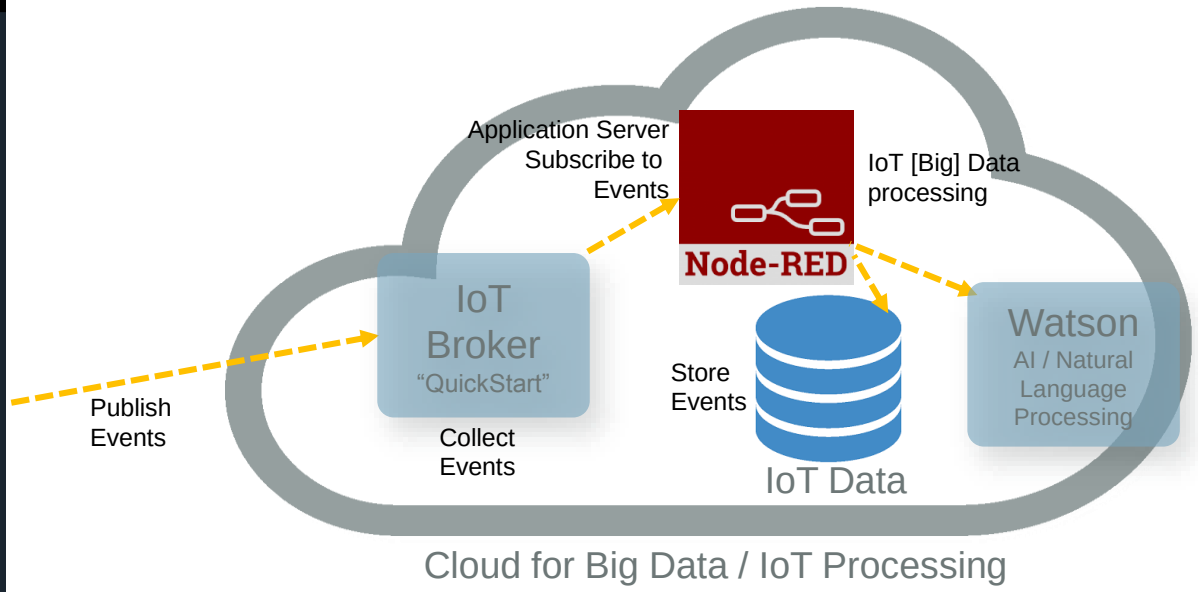


Lab 3 – Objectives & Architecture

- Create & modify an application using Node-RED
- Discover new services (IoT) & Node-RED, a visual tool (Open source project developed by IBM) to easily develop JavaScript applications, consume or create services (IoT / Watson…)



Virtual Temperature Sensor



IBM Client Center Montpellier

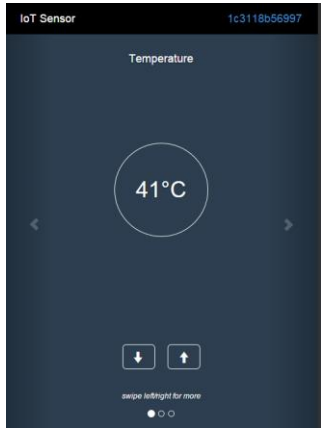
Lab 3 – Expected Results

Your Node-RED application is operationnal (using Node.js runtime), accessing Cloudant & IoT Foundation Services (QuickStart)

Your App is online (reachable from the Internet), & will be connected to a temperature simulator (sensor)

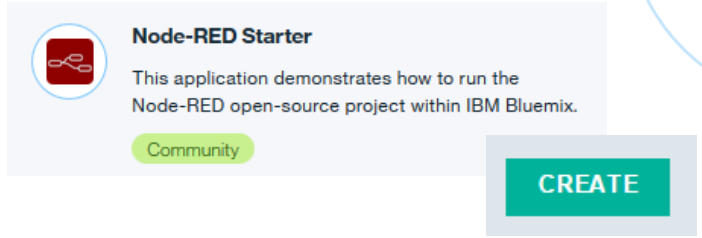
Prerequisites - Download the JSON file:

http://bluemix-watson-day.mybluemix.net/files/Lab3-bluemix-iot.json

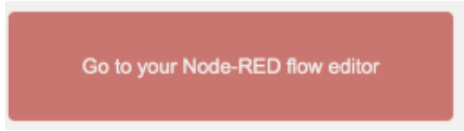


Lab 3 – IoT & Node-RED – Create and login

1. In IBM Cloud Catalog, choose “boilerplate” Node-RED Starter & 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.
   Click Create. Wait for the environment to be created & the App to start (~4 minutes).

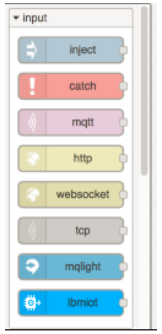


2. Access the Node-RED application (By clicking on the “Visit App URL”)
   Run the wizard to configure authentication: secure your editor with your own credentials so only authorized users can access it



NB: Don’t check “Allow anyone to view the editor, but not make any changes” and “Allow anyone to view the editor”

Have a look to the available IBM Cloud nodes

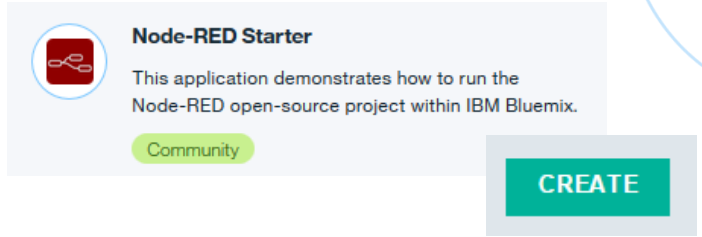


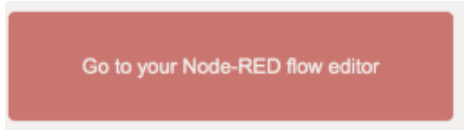
Click on Finish so start Node-RED will start

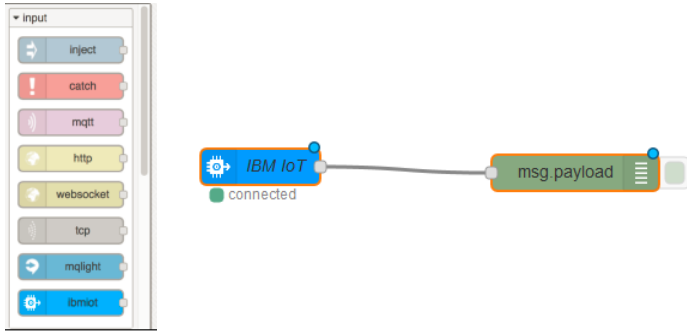
Click on Go to your Node-RED flow editor and use the credentials provided before

Lab 3 – IoT & Node-RED – Create a new Flow

3. Sensors & IoT – Create a simulator & identify your device ID (top right corner).
   Connect to **http://ibm.biz/iotsensor**
   Note: Instead of using your desktop browser, you can use your smartphone!
4. In the Flow Editor, Create a Flow (drag & drop of boxes on the left panel)
   Chose the Input node 'ibmiot' / « IBM IoT ». Add an output « Debug » node & link them.
5. Configure « IBM IoT » by double clicking on it :
   - Authentication: Quickstart (means it is a simple authentication – for demo purposes)
   - Device ID : <The value from Step 3 - Generated by the Simulator>
6. Click Done & Deploy your flow by clicking the « Deploy » button (top right).

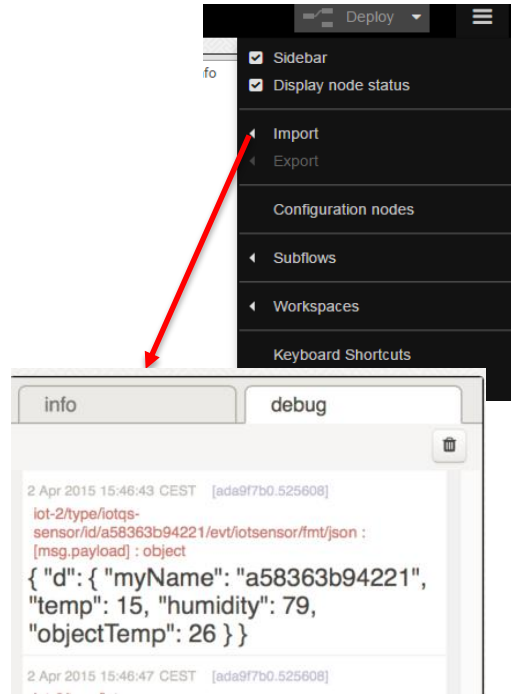


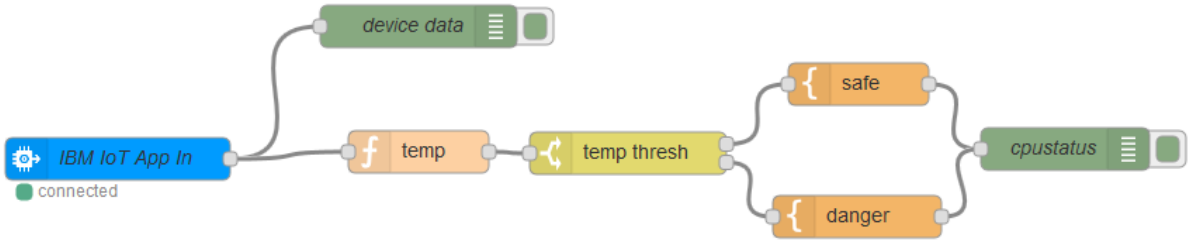




Lab 3 – IoT & Node-RED – Import a Flow

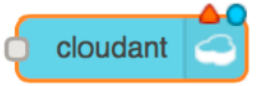
7. Check the Debug Panel on the right side while you are playing with the sensor simulator. You should receive Device (sensor) data as the IBM IoT Node subscribed to this particular Device topic.
8. Delete the whole Flow by selecting all the nodes & pressing the ‘Delete’ key.
9. Create a new flow – This time – by importing the code (URL: http://bluemix-watson-day.mybluemix.net/files/Lab3-bluemix-iot.json)
   - Click on the top right button near Deploy.
   - Select import, Clipboard & copy/paste the content of the JSON file
10. Fill in the Device ID field in the ‘IBM IoT App In’ Node.
    Deploy the new Flow.
    Modify the Device Temperature & check the Debug logs.



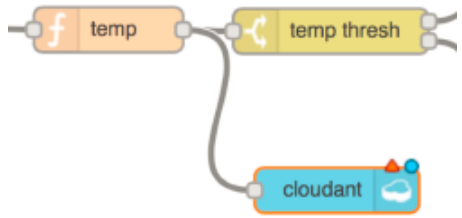


Lab 3 – IoT & Node-RED – Insert IoT Data in Cloudant DB

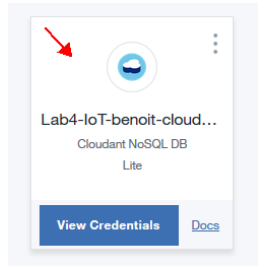
Let's insert the event data coming from the Device sensors in a cloudant Database!

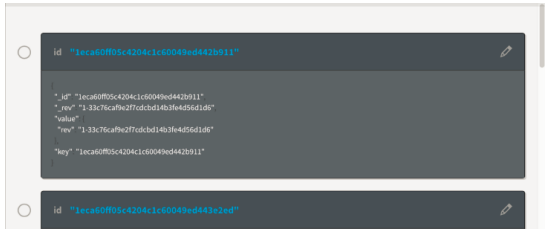


11. Add a Cloudant Node (Cloudant OUT node in the Storage Category) & link it to the « Temp » function node See picture on the right =>



12. Configure it:
    - 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
13. Deploy your new flow
14. From your IBM Cloud Dashboard, start the Cloudant console by clicking on the line of your NodeRed App (and not on the link), and have a look to the inserted data in the Database (name specified in step 12).





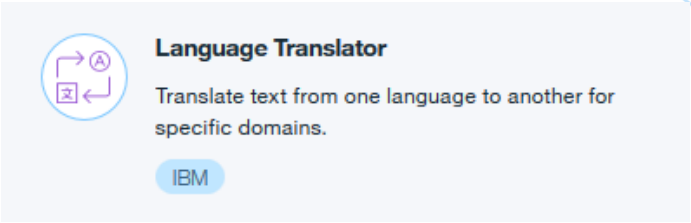
Lab 3 – IoT & Node-RED – Process IoT Data with Watson

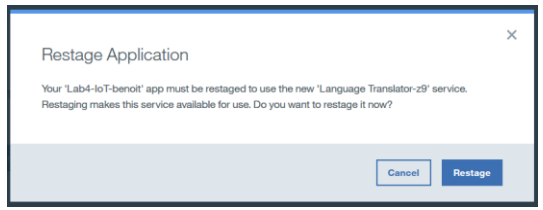
15. Add a « Watson Language Translator » service to your existing Node.js / Node-RED application and accept the Restage step to actually bind the service to the app.
    > App IBM Cloud Dashboard > Connections > Connect New, then click CREATE

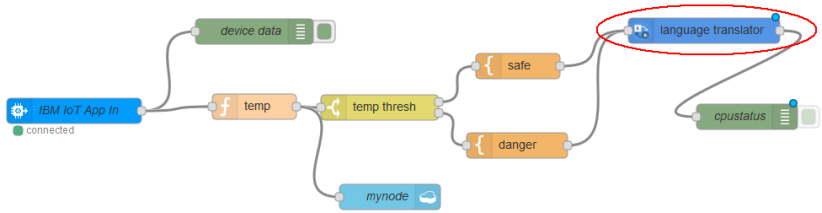
    Note: while it is restaging, go to Credentials : This information are useful if you want to invoke your Watson Service from any program (running in IBM Cloud or outside IBM Cloud)

16. In Node-RED, add a 'languagetranslator' / « Watson language translation » node and link it between the template (*safe & danger) & debug cpu status* nodes. Configure the Watson language translator node:
    - Name (of your choice)
    - Mode: keep "translate"
    - Domains: Conversational
    - Source: English
    - Target: French (or Spanish, Portuguese & Arabic)
    - Note: The user/password fields are not necessary & do not appear in the node settings if a Watson Translator service is properly bound to Node.js !!!

17. Deploy your flow & check the logs!!







End of Lab


Thank You


Benoît Marolleau
IBM Certified Experienced Architect
Cloud Computing
Power Systems
IBM
IBM Client Center Montpellier
Parc Industriel de la Pompignane
34000 Montpellier
Phone: +33 4 67 34 63 35
benoit.marolleau@fr.ibm.com
IBM

To go further…

- Reference
  - https://github.com/apischdo/WOW2016
    - IBM Watson Visual Recognition APIs
    - Advanced dialog with Conversation….
    - More to come…
- Watson Developer Cloud (Docs, Demos, Tutorials…)
  - http://www.ibm.com/watson/developercloud/