



Bluemix Hands On Labs Series

Watson on Node-RED



IBM France Lab

Version:

1

Last modification date:

24th May 2016

Owner:

Yves LE CLEACH

Node-RED in Bluemix

Node-RED in Bluemix

A visual tool for wiring the Internet of Things with Watson



This is a fork of the Node-RED in Bluemix boilerplate that integrates the following :

- All available Watson nodes for Node-RED on [GitHub](#)
- Flows from the [Watson Node-RED Basics Labs](#)
- added Dropbox and Box nodes in the palette

This Boilerplate shows basics flows sample of the Watson nodes :

[Go to your Node-RED flow editor](#)

[Learn how to password-protect your instance](#)

[Learn how to customise Node-RED](#)



Table of Contents

Bluemix Hands On Labs Series.....	1
Watson on Node-RED	1
1 Introduction.....	4
2 Lab Objectives.....	5
3 Pre-requisites.....	5
4 Discovering Node-RED	6
4.1 Create your first Node-RED application on Bluemix.....	6
4.2 Creating your first flow	8
4.3 Importing or Exporting Flows	9
4.4 Securing your Node-RED editor.....	11
5 Discovering Watson on Node-RED	12
5.1 Watson Node-RED Starter.....	12
5.2 Basics Labs	17
5.2.1 Watson Language Translation and Identification service.....	17
5.2.2 Watson Dialog service	18
5.2.3 Watson Tone Analyser service	19
5.2.4 Alchemy Image Analysis	19
6 Advanced AI Demo using Watson NLC	20
7 Additional Resources	21
8 Annexes.....	22
8.1 Create your Bluemix account	22
8.2 Create your JazzHub account.....	22

1 Introduction

[Node-RED](#) is an open-source prototyping tool that uses a drag/drop/wiring paradigm to build applications. Complex applications can be built quickly with little or no programming required and enable you to connect together hardware devices, APIs and online services in new and interesting ways. It is open source, and can be installed on small devices (e.g. raspberry pi), but a customized version is proposed on Bluemix. (different set of nodes than the local version).

Node-RED offers a large palettes of ready-to-use nodes. The Watson nodes presented in this Lab are a subset of the Cognitive APIs available through the Watson Developer Cloud.

IBM Watson Developer Cloud is a set of micro-services available as REST APIs that applications can use to incorporate cognitive capabilities. Credential access to the APIs is available through Bluemix - IBM's cloud platform as a service.

This document is based on a collection of Basic and Advanced Labs available on Git :

<https://github.com/watson-developer-cloud/node-red-labs>

The basic labs are simple, standalone examples developed to show how to call each individual Watson Node-RED node. The advanced labs build on this to bring multiple nodes together to create even more complex applications.

The intent of this Lab document is to have an introduction to Node-RED and Watson nodes in a 2 hours session format.

Important : all the presented nodes are open-source, so do not hesitate to send feedbacks through GitHub website (bugs, documentation issue, feature requests), or any suggestion to this document to yves.lecleach@fr.ibm.com.

2 Lab Objectives

- Discover Node-RED and learn essentials development tips.
- Discover some of the Watson API or Alchemy API through the Watson nodes on Bluemix
- Build a Cognitive Node-RED application using Watson nodes

3 Pre-requisites

For the following Lab exercices, you will need the following:

- a Bluemix Account : cf Annexes.
- a JazzHub account : cf Annexes.
- for Labs that are using the Watson APIs : all APIs keys are provided and managed by Bluemix.
- for Labs that are using the Alchemy APIs you will need an API key (ask the instructor)
- (optional for AI Lab section 6) : Install Cloud Foundry Client on your laptop :
<https://github.com/cloudfoundry/cli#downloads>

4 Discovering Node-RED

4.1 Create your first Node-RED application on Bluemix

First, log in onto your Bluemix account : <https://bluemix.net> and creates a Space ‘HandsOnLabs’ in which you will work for this lab session.

Then, go to the Catalog and click on the Node-RED Starter in the Boilerplates category:



In the Name field, type “nodered-XXX” (replace XXX by your initial, so the hostname is unique) and click on the Create button.

The screenshot shows the Bluemix Catalog interface. On the left, the Node-RED Starter entry is displayed with its icon, name, version (0.4.25), type (Boilerplate), and a 'VIEW DOCS' button. In the center, there's a detailed view of the Node-RED Starter, including its description: "Develop, deploy, and scale server-side JavaScript® apps with ease. The IBM SDK for Node.js™ provides enhanced performance, security, and serviceability.", and a 'VIEW DOCS' button. To the right, a 'Create an app:' dialog is open, prompting for 'Space' (HandsOnLab), 'Name' (nodered-XXX, highlighted in blue), 'Host' (nodered-XXX), and 'Domain' (mybluemix.net). Under 'Selected Plan:', 'SDK for Node.js™' is chosen with 'Default' selected, and 'Cloudant NoSQL DB' is set to 'Shared'. A 'CREATE' button is at the bottom of the dialog.

The provisioning of your new node-RED instance is starting...

After 2-3 min you can start developing in your node-red app, entirely in the cloud.

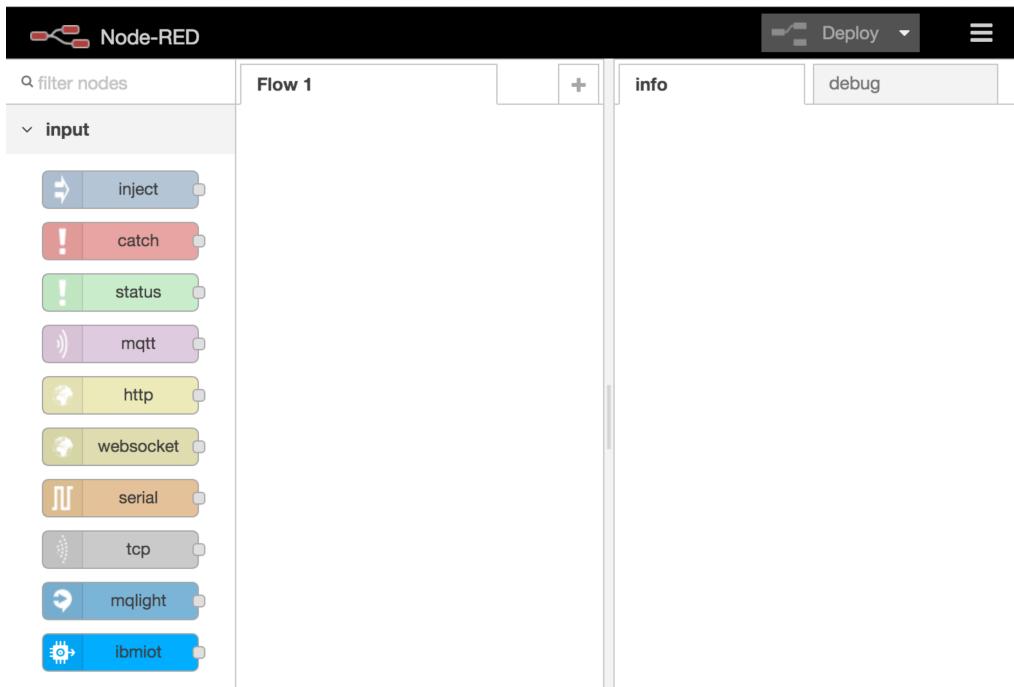
In a Node-RED app, which is run by a Node.js server instance, you can define any numbers of functionality you want, as on any application servers. So a Node-RED app, can host any numbers of applications you desire, as on a JEE application server. (e.g: Liberty Java server). In another word: a node-RED app is an application server that you can configure in Bluemix easily to add multiple instances, increase memory, monitor, manage. Etc...

The screenshot shows the Bluemix app dashboard for the 'nodered-YLC' application. At the top, it displays the app name and a 'Routes' link. Below that is a configuration section with 'INSTANCES' set to 1, 'MEMORY QUOTA:' at 512 MB, and 'AVAILABLE MEMORY:' at 2.000 GB. There are 'SAVE' and 'RESET' buttons. To the right, there's an 'APP HEALTH' section indicating the app is running, and a 'RESTART' or 'STOP' button. The 'ACTIVITY LOG' section shows three entries from May 23, 2016, at 3:11 PM, made by 'yves.lecleach@fr.ibm.com': starting the app, updating it, and creating it. At the bottom, there's a link to 'Estimate the cost of this app'.

Click on the Routes link, you should arrive on this default page:

This is the default landing page for the Node-RED application in Bluemix. It features a dark header with the text 'Node-RED in BlueMix'. The main content area has a light gray background. At the top, it says 'Node-RED in BlueMix' and 'A visual tool for wiring the Internet of Things'. Below this is a diagram of a node flow with several nodes connected by lines. A descriptive text block on the left says: 'Node-RED provides a browser-based editor that makes it easy to wire together flows that can be deployed to the runtime in a single-click.' On the right, a red button-like callout box contains the text 'Go to your Node-RED flow editor'.

Now click on the RED button to enter the web-based node-RED development tool :



You can now start developing your node-red applications.

4.2 Creating your first flow

We will create a Hello World flow.

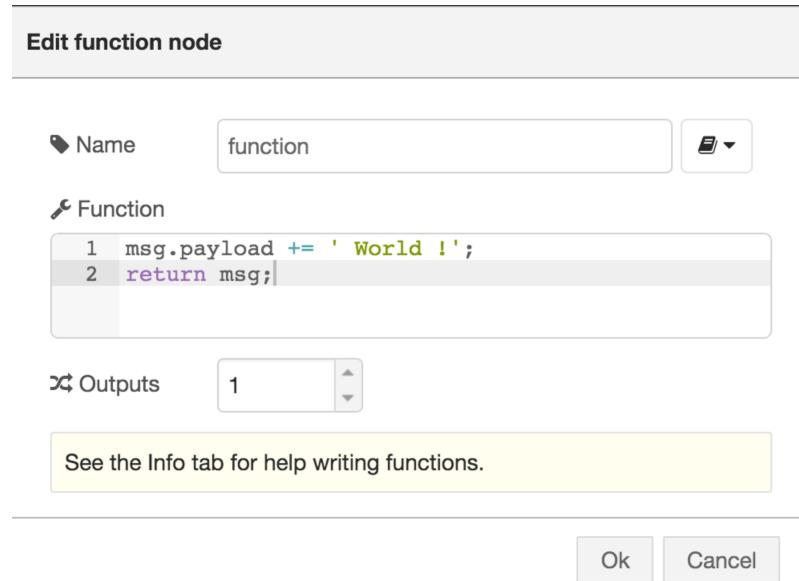


This program is a very simple flow that prints the message 'Hello World' on the screen. Here you can see Node-RED's user interface, the colored blocks on the screen are called nodes, which is a visual representation of a piece of JavaScript code to carry out a task. To build this 'Hello World' flow you need to take the following steps:

1. Drag an 'Inject node' to the canvas
2. Double click this node to see the options
3. Use the drop-down, to select string for the payload
4. Type 'Hello' on the second line: This will cause to inject hello into the flow when clicked on the inject node) and click on ok, to save and close this node.
5. Add a 'Function node' to the canvas, open it and place this on the first line into the function:

```
msg.payload += ' World !';
```

This will add 'World !' to the payload. The complete function should look like this:



6. Add a 'Debug node' to the canvas.
7. Wire the 'Inject node' to the 'Function node' and the function node to the 'Debug node'. Most nodes have a grey circle on their left side, which is their input port, and on their right side, which is their output port. Left clicking and dragging the output to the input port of the next node connects the two together.
8. Press 'Deploy'.

Now you have build your first Hello World flow. Test it by clicking on the 'Inject node', you will see some output in the debug window on the right (click on 'Debug' to change the view from info to debug).



4.3 Importing or Exporting Flows

With Node-RED you can export/import easily the flows you have designed or ones from the node-red community. (<http://flows.nodered.org/>)

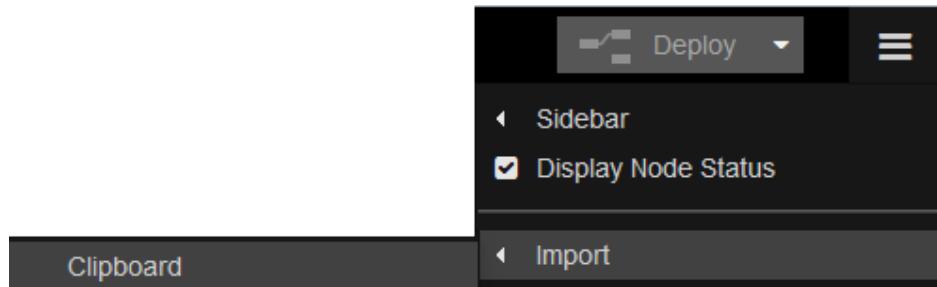
If you want to try this now then select the sample flow below

https://github.com/watson-developer-cloud/node-red-labs/blob/master/introduction_to_node_red/import_sample_flow_flow.json

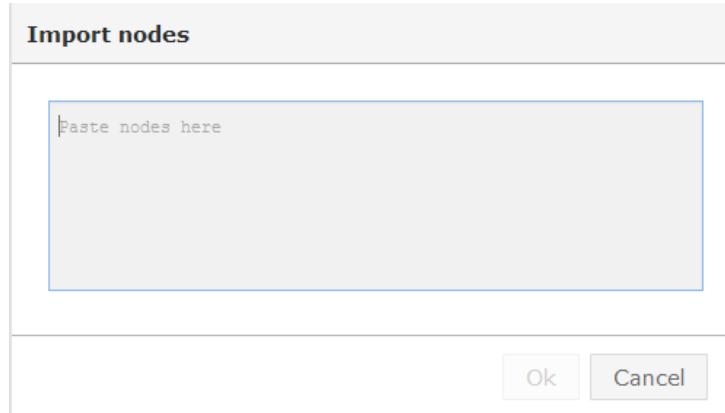
Select the flow and copy it to the clipboard (Ctrl-C). Import the flow into Node-RED using by selecting the node-RED menu



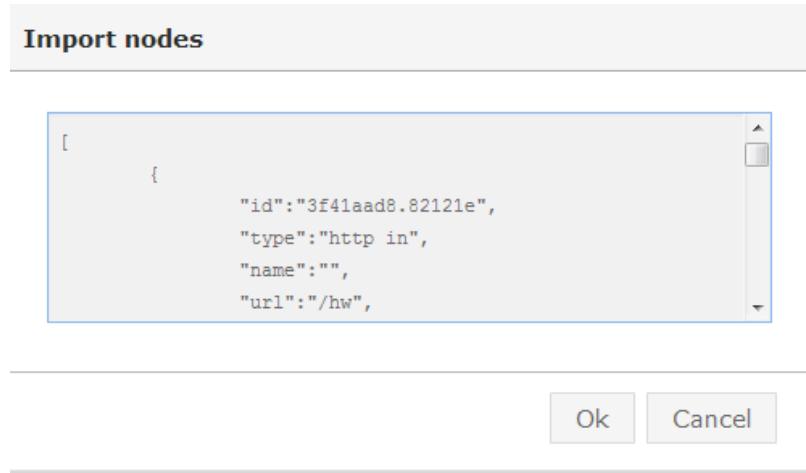
and select the import from clipboard option.



You will be presented with a form in which you create nodes by entering json data.



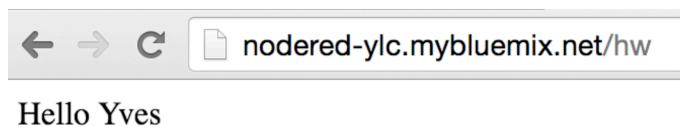
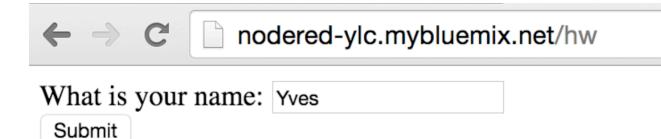
Import your copied flow by pasting (Ctrl-P) from the clipboard into the the form.



Place the imported flow onto your node-RED page and press DEPLOY. (this button Deploy and save your work in a Cloudant instance).



You can test by entering this URL : (replace the hostname by yours)



4.4 Securing your Node-RED editor

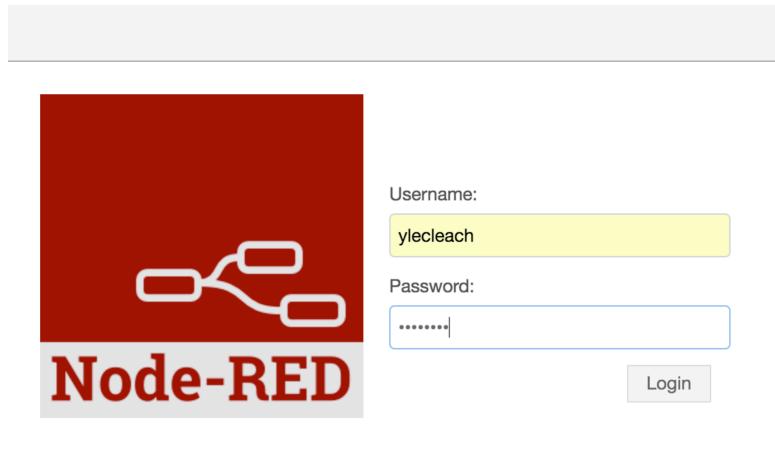
By default, your Node-RED editor will be public, let's configure the access: On the left pane menu, select the 'Environment Variables' page

Add the following USER-DEFINED variables:

- NODE_RED_USERNAME - the username to secure the editor with
 - NODE_RED_PASSWORD - the password to secure the editor with

And Click on the Save button. Your Node-RED app will restage quickly.

Now, when someone will click on the button of the Node-RED editor, an authentication dialog box will be prompted.



5 Discovering Watson on Node-RED

5.1 Watson Node-RED Starter

This section aims to clone and deploy in Bluemix the new Watson Node-RED Starter application that will help your to quickly test the following Basics or Advanced Watson Labs using Node-RED.

This fork of the Node-RED Starter in Bluemix integrates the following :

- All available Watson nodes for Node-RED on [GitHub](#)
 - <https://github.com/watson-developer-cloud/node-red-node-watson>
- Flows from the [Watson Node-RED Basics Labs](#)
 - https://github.com/watson-developer-cloud/node-red-labs/tree/master/basic_examples
- Added Dropbox and Box nodes in the palette

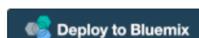
Lab Instructions : Now Install the Watson Node-RED Starter by clicking on the Deploy button :
<https://github.com/watson-developer-cloud/node-red-bluemix-starter>

README.md

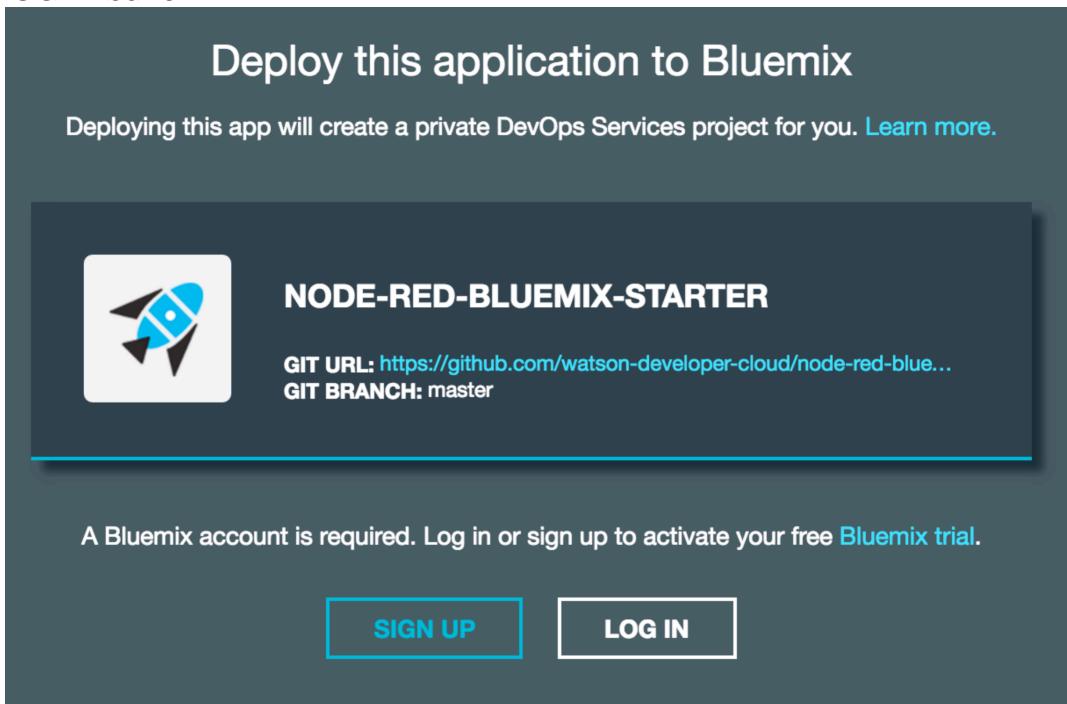
Node-RED Watson Bluemix Starter Application

Node-RED Watson in Bluemix

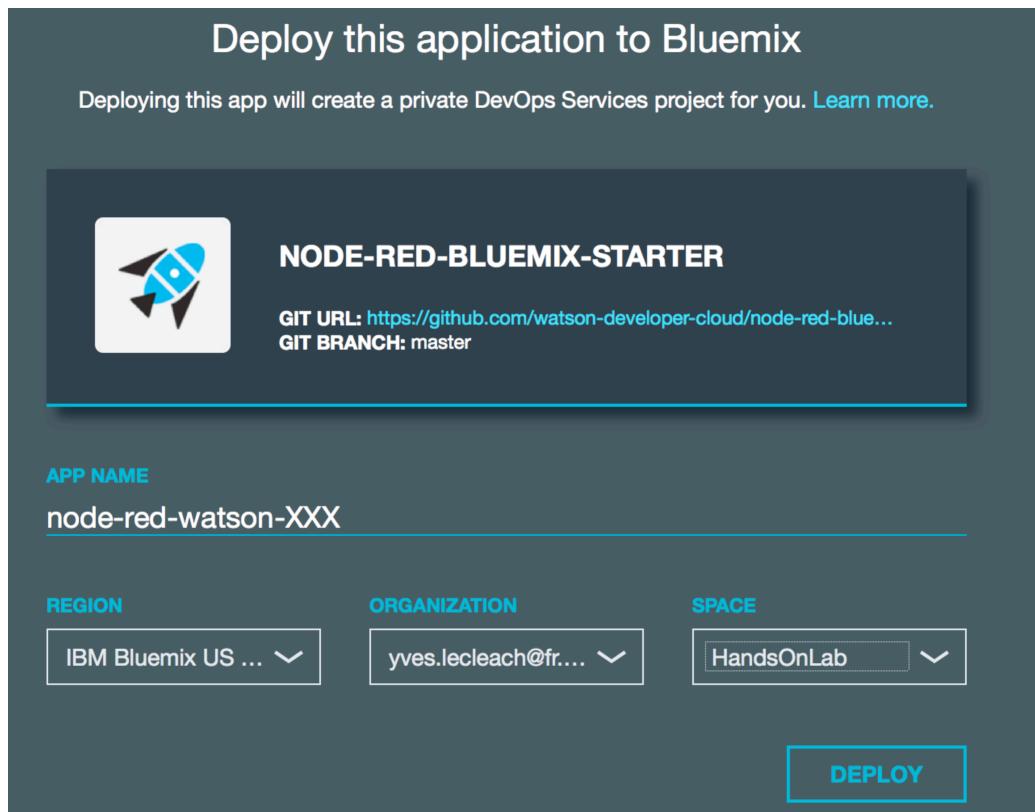
This repository is an example Node-RED application that can be deployed into Bluemix with only a couple clicks. Try it out for yourself right now by clicking:



Click on the LOG IN button :



Then enter a unique application name as usual on Bluemix :



Then click on the Deploy button.



After 2-3 min the application should be ready. You will receive an email that explains that a project has been created in your JazzHub account.

You can customize the project if you wish to do so. (not covered in this Lab)

Now click on your app icon, and open the app through the Routes link :

You should have this welcome page :



Node-RED in Bluemix

A visual tool for wiring the Internet of Things with Watson



This is a fork of the Node-RED in Bluemix boilerplate that integrates the following :

- All available Watson nodes for Node-RED on [GitHub](#)
- Flows from the [Watson Node-RED Basics Labs](#)
- added Dropbox and Box nodes in the palette

[Go to your Node-RED flow editor](#)

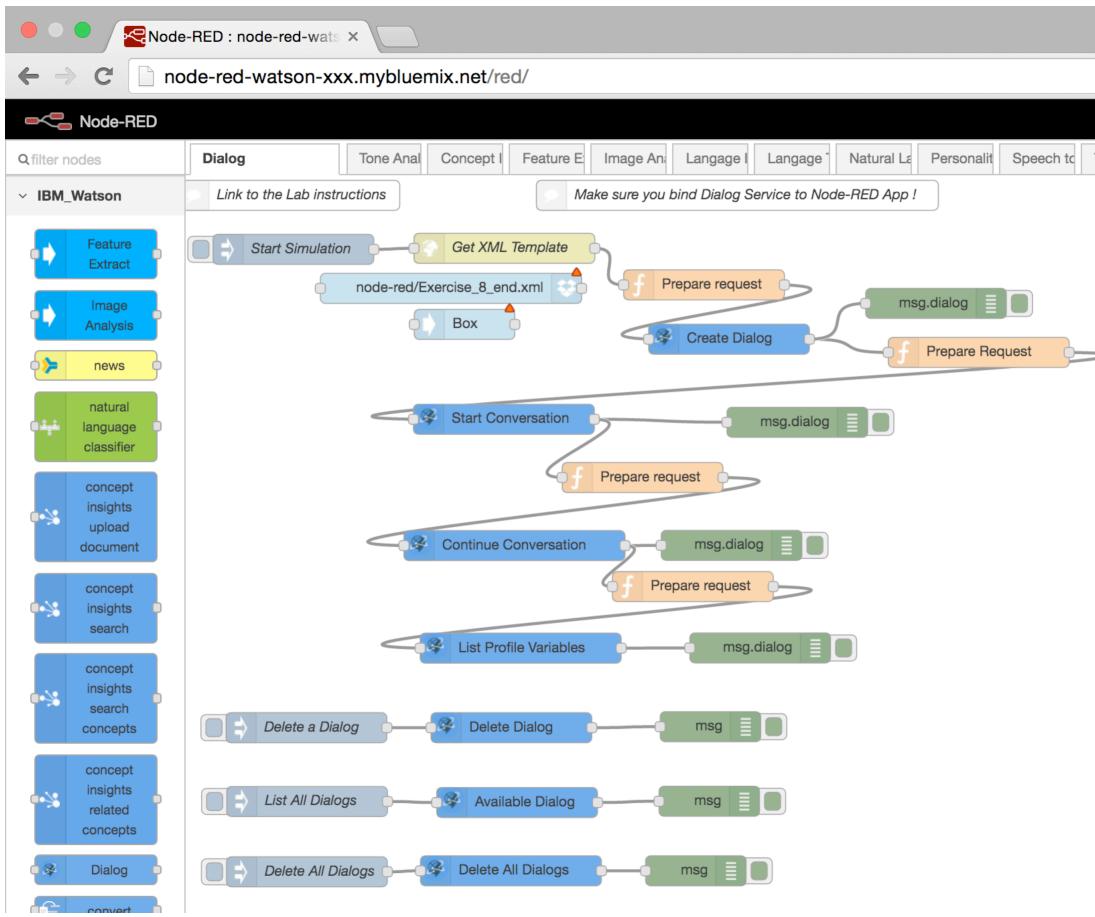
[Learn how to password-protect your instance](#)

[Learn how to customise Node-RED](#)

This Boilerplate shows basics flows sample of the Watson nodes :

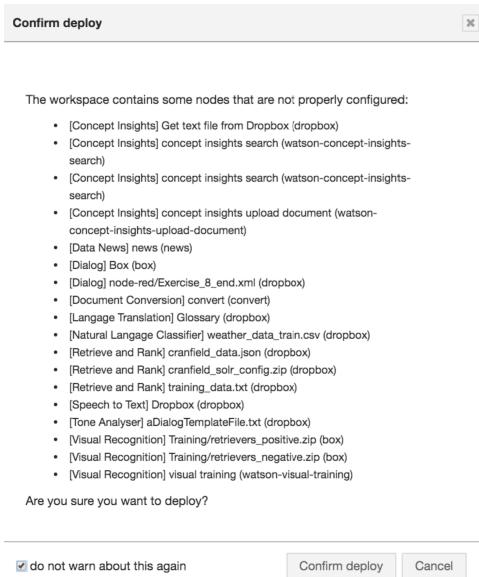
- Alchemy Image Analysis
- Alchemy Data News
- Alchemy Feature Extraction
- Dialog Template Creation
- Dialog
- Document Conversion

To start discover the Watson APIs through the Watson nodes, just click on the red button. The first tab is opened by default and contains the final Flow of the Dialog Lab that will be done in section 5.2.2.



About this Watson node-red Starter :

- this app contains all the flows described at the end of each Basic Labs and is intended for educational purposes.
- The Link to the Lab Instruction : click on the info tab at the right of the Node-RED editor to see the link of the corresponding Lab.
- You have to instantiate the corresponding Watson service in your bluemix, in the same space of this application.
- Ignore the Warning dialog box by selecting “do not warn about this again”



Notice : the Node-RED Starter in Bluemix catalog also contains the Watson codes from GitHub. (since May 2016)

5.2 Basics Labs

The basics labs will show you how you can use the Watson nodes in Node-RED.

You can either discover by yourself the Basic Labs, or follow the selection in section 5.2.1.

https://github.com/watson-developer-cloud/node-red-labs/blob/master/basic_examples/README.md

Some Watson APIs are very simple to use, others are more complex with a training mode that must be done prior using the API. Some others as Watson Language Translation service offers pre-defined domain specifics domains.

5.2.1 Watson Language Translation and Identification service

This API can be used either without a training mode or with a training mode. As an example, the training mode of this service can be used to define language translation in a particular business domain. (e.g : in a car manufacturing domain, in a medical domain, ...). The service is configured by default with a News domain, a Conversational domain (colloquialisms) and a Patent domain.

Notice that Watson Language Translation supports translation to some Regional languages.

Translate mode

Follow the first part of the Lab, to test translation from EN to FR :

https://github.com/watson-developer-cloud/node-red-labs/tree/master/basic_examples/language_translation

Customizing your domain mode

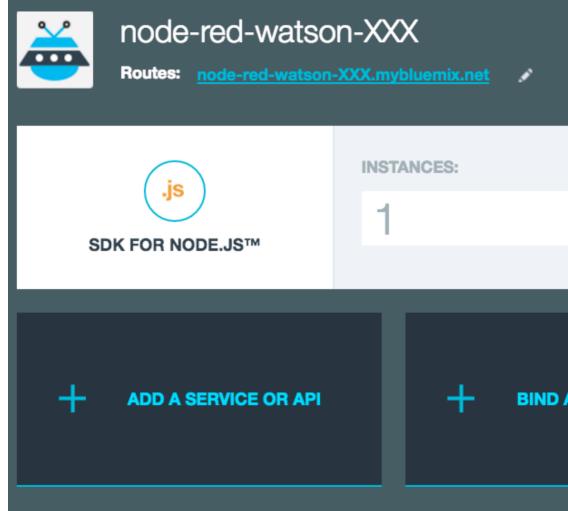
Follow the second part of this Lab, to try the Training of this service with a customize domain. (provided)

Language Identification

Follow the following Lab :

https://github.com/watson-developer-cloud/node-red-labs/tree/master/basic_examples/language_identification

Notice : do not forget to add a Watson Language Translation service instance to your app with the ‘Trainable’ plan using the ADD A SERVICE OR API. A restage of the app is required.



Watson documentation :

<https://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/doc/language-translation/>

Watson API documentation :

<https://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/language-translation/api/v2/>

5.2.2 Watson Dialog service

The IBM Watson Dialog Service allows a developer to design the way an application interacts with an end user through a conversational interface. The Dialog service enables applications to use natural language to automatically respond to user questions, cross-sell and up-sell, walk users through processes or applications, or even hand-hold users through difficult tasks. The Dialog service can track and store user profile information to learn more about end users, guide them through processes based on their unique situation, or pass their information to a back-end system to help them take action and get the help they need.

Follow the following Lab or directly execute it on you Watson node-RED app :

https://github.com/watson-developer-cloud/node-red-labs/tree/master/basic_examples/dialog

Notice : do not forget to add a new Watson Dialog service instance to your app using the ADD A SERVICE OR API. A restage of the app is required.

Watson documentation :

<http://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/doc/dialog/>

Watson API documentation :

<http://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/dialog/api/v1/>

5.2.3 Watson Tone Analyser service

People show various tones, such as joy, sadness, anger, and agreeableness, in daily communications. Such tones can impact the effectiveness of communication in different contexts. Tone Analyzer leverages cognitive linguistic analysis to identify a variety of tones at both the sentence and document level. This insight can then be used to refine and improve communications. It detects three types of tones, including emotion (anger, disgust, fear, joy and sadness), social propensities (openness, conscientiousness, extroversion, agreeableness, and emotional range), and language styles (analytical, confident and tentative) from text.

Follow the following Lab or directly execute it on your Watson node-RED app :

https://github.com/watson-developer-cloud/node-red-labs/tree/master/basic_examples/tone_analyser

Notice : do not forget to add a new Watson Tone Analyser service instance to your app using the ADD A SERVICE OR API. A restage of the app is required.

Watson documentation :

<https://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/doc/tone-analyzer/>

Watson API documentation :

<https://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/tone-analyzer/api/v3/>

5.2.4 Alchemy Image Analysis

Using the Image Analysis node, you can use the Alchemy APIs to analyse images for face detection, content tags and links.

The following features are available for analysis:

- **Faces**, discover human faces within the image.
- **URLs**, extract images from a URL webpage.
- **Content Tags**, extract content tags from the image.
- **Text**, extract text from the image.

For full details on the feature details and response values, please see the [Alchemy API documentation](#)

Follow the following Lab or directly execute it on your Watson node-RED app :

https://github.com/watson-developer-cloud/node-red-labs/tree/master/basic_examples/alchemy_api_image_analysis

Bonus : Apply Content Tags on a Caricature.

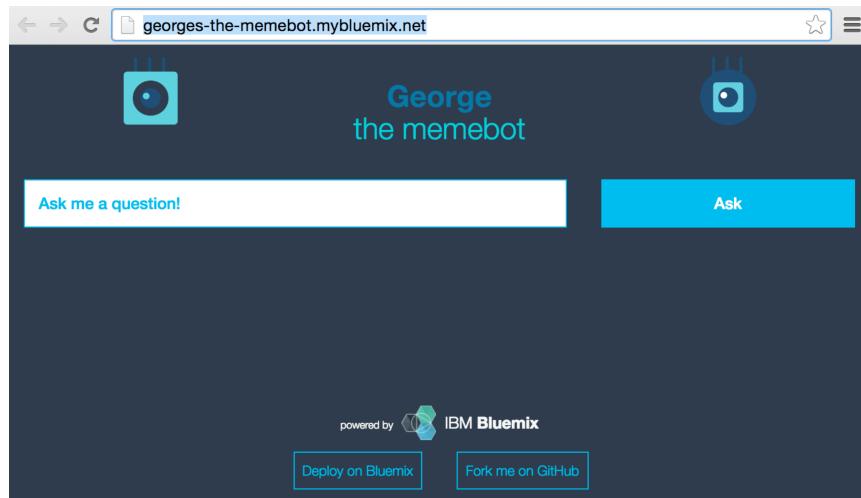
Import this flow in the same node-red editor pane, configure the API key in the node, and run it :

<https://raw.githubusercontent.com/ylecleach/misc/master/caricature-flow.json>

Notice : to use this node, configure the API key provided by Alchemy or your instructor.

6 Advanced AI Demo using Watson NLC

This demo is a Node.js demo based on the Watson Natural Language Classifier. It can be done without a prior knowledge of Node.js. To Clone Georges, all you have to do is to click on the “Deploy on Bluemix” button on the following URL : <https://georges-the-memebot.mybluemix.net>



You will have then to Train your service as described in the Lab available on my github; See “How to Train your Robot”:

<https://github.com/ylecleach/george-the-memebot>

Ask any existential questions to Georges, He will be happy to answer you by replying you with a Meme.

What is a Meme ? see <https://en.wikipedia.org/wiki/Meme>

Conclusion

What next ? modify Georges with additional Watson services so Georges baby AI can grow up to become a human-class and world-wide artificial intelligence. Add in its gene some Watson Dialog and Retrieve and Rank service, and spread it over the Internet. Are you ready ?

7 Additional Resources

Reference Node-RED Nodes: <https://github.com/watson-developer-cloud/node-red-node-watson>

Node-RED Labs: <https://github.com/watson-developer-cloud/node-red-labs>

Node-RED Bluemix Starter: <https://github.com/watson-developer-cloud/node-red-bluemix-starter>

Node-RED support :

- Introduction to Node-RED: <https://ibm.biz/BdHvfc>
- [Node Red Forum on GoogleGroups](#)

Watson on Node-RED support :

- Watson Node-RED Starter kit: <https://ibm.biz/Bd4nCJ>
- Basic Labs: <https://ibm.biz/BdHvfD>
- Advanced Labs: <https://ibm.biz/BdHvfX>
- Contact committers on GitHub.

You can find additional resources on Watson :

- [Watson Developer Cloud](#) : main documentation on Watson APIs
- [Watson API Explorer](#) : useful for developers to discover Watson REST API
- [Watson Developer Cloud Gallery](#) : many watson nice demos
- [Watson Cloud SDK for Node](#)

Want more, Want to share your Bluemix or Watson experiences in France

Bluemix communities in France :

- Bluemix Paris Meetup : <http://www.meetup.com/IBM-Bluemix-Paris-Meetup/>
- Bluemix Usergroup France blog : <http://usergroup.bluemix.fr>

Follow us :

- Twitter : <https://twitter.com/IBMFranceLab>
- Twitter : <https://twitter.com/ylecleach>
- GitHub : <https://github.com/ylecleach>

Annexes

7.1 Create your Bluemix account

IBM Bluemix is a cloud Platform as a service (PaaS) which supports several programming languages (Java, Node.js, Go, PHP, Python, Ruby...) and services as well as an integrated software development method called **DevOps** to build, run, deploy and manage applications easily on the cloud. Bluemix is based on Cloud Foundry open technology and runs on a SoftLayer infrastructure.

*If you already have a Bluemix account, connect with your **IBM ID** and your Bluemix password. If not, please follow the instructions.*

1. _ Connect to <https://bluemix.net> and click on “Get started free”



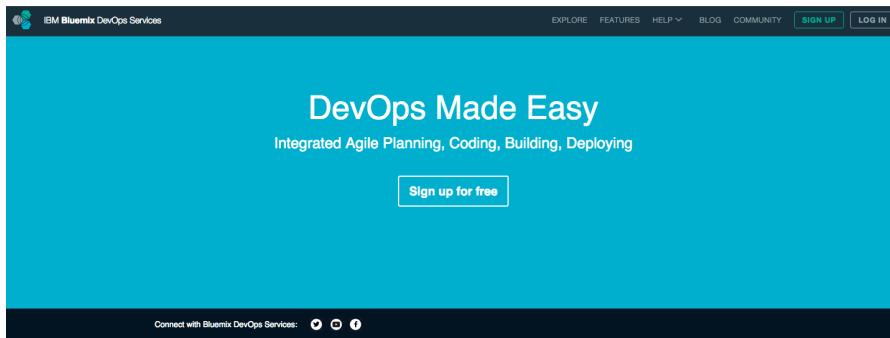
2. _ If you don't have an **IBM ID** yet, please fill in the form and click on “Create account”, it will create an **IBM ID** and a Bluemix account. By default the Bluemix account will provide you free 30-day trial access with no credit card required. You get access to:

- 2 GB of runtime and container memory to run your applications,
- unlimited IBM services and APIs (max 10 services instances)
- and complimentary support.

7.2 Create your JazzHub account

Please follow the 3 steps below.

1. Browse <https://hub.jazz.net/> and click on SIGN UP (up and right).



IBM DevOps approach is a set of practices, tools and services for accelerating the development and release cycle of software applications. It helps enterprises get their ideas into production fast.

2. Then, click on “Log in to start using DevOps Services”

A screenshot of the "Sign Up" form. The title "Sign Up" is at the top. There are three numbered steps: 1. Create an IBM id (which is highlighted in blue), 2. Log in, and 3. Pick an alias. Below the steps, there's a note: "Already have an IBM id? [Log in to start using DevOps Services.](#)". The form fields include: Email address (IBM id), First name, Last name, Create password, Re-enter password, Country of Residence, and Affiliation. There's also a section for security questions: Security question and Security question answer. A checkbox for opting in to receive information from IBM companies worldwide is present. At the bottom, there's a link to accept the Terms of Use and two buttons: "Next" and "Cancel".

3. _ Log In and chose an alias associated to your **IBM ID**. The alias is unique and it will be used as a short name in Git repository paths. Click on “Continue”. You successfully registered on DevOps Services.