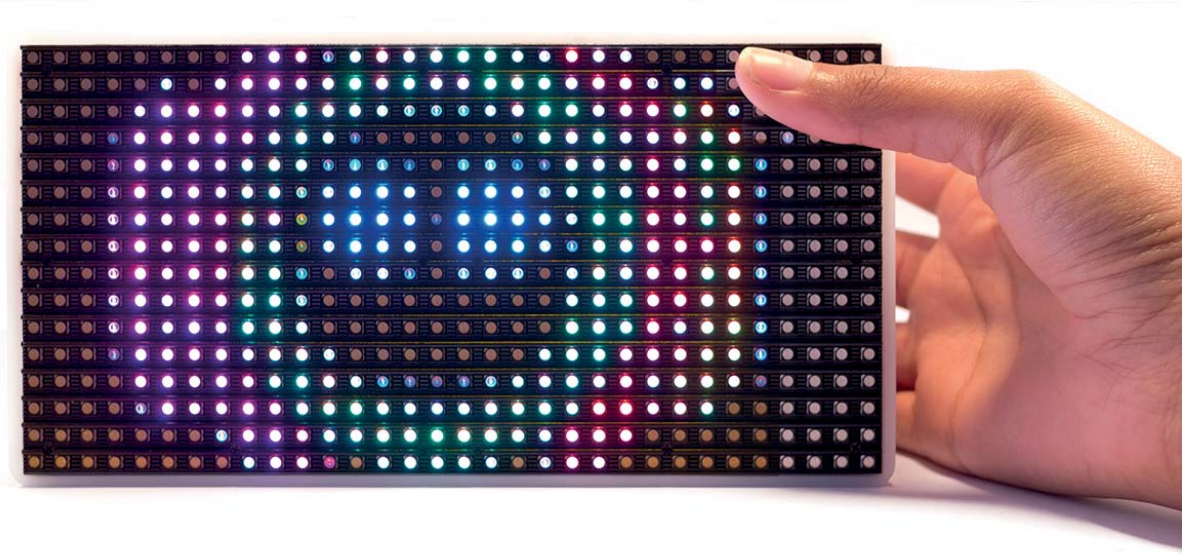
TimeSquAir



User Guide

Support

Any question? Any idea?

Use the form on the website **http://TimeSquAir.io**

Get additional documentation

http://thethingbox.io/docs/TSA-Book.pdf

Welcome

What is TimeSquAir, what is it for?

You have just purchased a TimeSquAir, the smart connected display. Its use is easy and this installation guide is here to help you step by step through the installation, commissioning and first uses of your TimeSquAir.

No technical knowledge is required. Everything is done so that whatever your age and knowledge, you will have fun creating cool stuffs with TimeSquAir.

TimeSquAir has the four most used by objects connected services:

- Displays your information on its LED matrix
- Plays sounds and music (by adding speakers)
- Reads NFC tags
- Listens internet and is searching the information it receives to give you those that interest you most.

All these services are managed using a visual editor.

Unpackage

Make sure you have all the following items in your box:

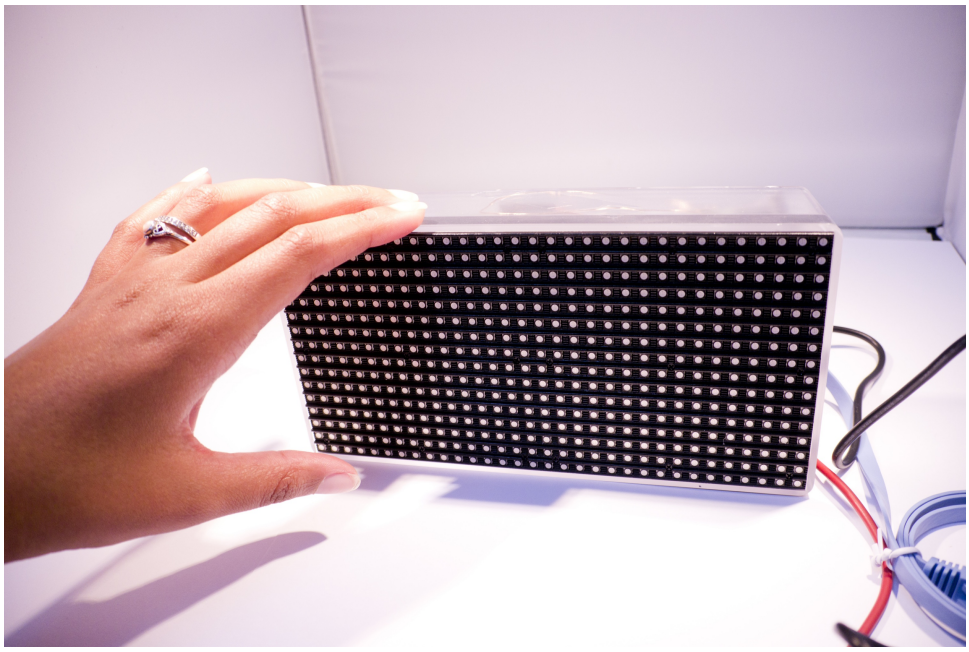
- A TimeSquair
- This book
- NFC tags
- An ethernet adaptator

Plug

You'll plug in and start TimeSquAir for the first time. It will take approximately three minutes.

How to install TimeSquAir?

TimeSquAir displays information from left to right, in the horizontal direction. Make sure to lay it horizontally on a stable surface.



Plug-in TimeSquAir :

- Connect the Ethernet cable to your Internet box.
- Plug TimeSquAir in an electrical outlet.
- Let TimeSquAir launch ...

TimeSquAir has a third red jack wire for audio, needed to play music, the radio or to talk to you.It also has several additional USB ports (open TimeSquAir to see and use them).

Start TimeSquAir for the first time

Immediately after the TimeSquAir connection on a socket, you will see (in that order):

- **TimeSquAir's logo** that scrolls for a few seconds. This means that TimeSquAir's software starts.
- **A message displaying the IP address** (a series of numbers with dots between them) : this means that TimeSquAir found its network and is ready to use.

If you do not have these three indicators at the beginning, this means that TimeSquAir has encountered a problem. See **http:/TimeSquAir.io**, "Help" at the bottom of the page for more information.

Where to go now

Now its time to plug and launch TimeSquAir!

Use the dedicated "**First Launch**" tutorial.

First Launch

This tutorial explains the registration process of the first launch of TimeSquAir.

Access with a browser

All you need is a computer on the same local network to manage your TimeSquAir, using an Internet browser.

Use your favorite (Google Chrome and Mozilla Firefox are the most tested).

Enter:

```
http://timesquair.local/
```

in the address bar.

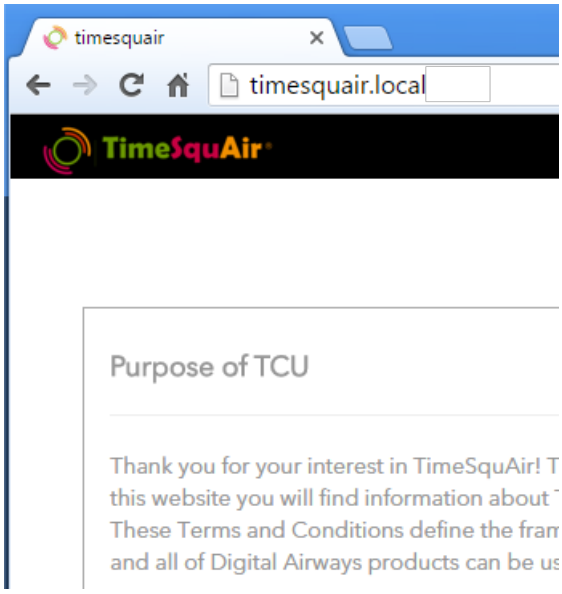
The last slash may be ommitted in some browsers.

If this does not work:

- make sure you are on the same local network
- be aware that some electronic router may block that address
- Try the IP address (the four dot separated numbers at are displayed at the launch time) something like `http://192.168.0.23`

Register

Now you get the usual boring legal notices that you should of course accept.

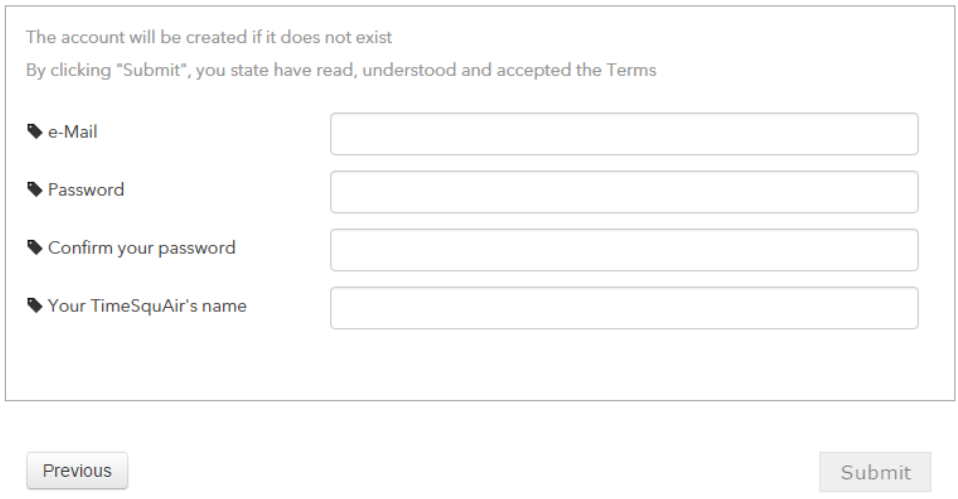


Let's go registering!

The account has two purposes:

- first it allows us to communicate with you, sending news and (most important) TimeSquAir updates.
- second, TimeSquAir uses it as an identifier for cloud services

The account is the **only** way for us to get your email and stay in touch with you: fill is seriously (and read the inbox!).



Last but not least, you have to change the network name of your TimeSquAir. Use names as "PaulTSA", "OfficeTSA"...

Beware that the address to use in the browser address bar changes according to the new network name!

If your new TimeSquAir name is "PaulTSA", use:

```
paultsa.local/
```

To access it.

Where to go now

- Use the tutorial "**Try and understand**" to see some magic without waiting any longer and understand the underlying behavior of TimeSquAir
- Use the tutorial "**The Visual Editor**" to discover IBM's Node-RED that allows to change the behavior of your TimeSquAir.
- Is an update available ? Use the "**Update TimeSquAir**" tutorial.

Try and understand

This tutorial explains the basic behavior of TimeSquAir, using a very short typical example.

The top of TimeSquAir contains a tag reader.

Let's try it as a first TimeSquAir use.

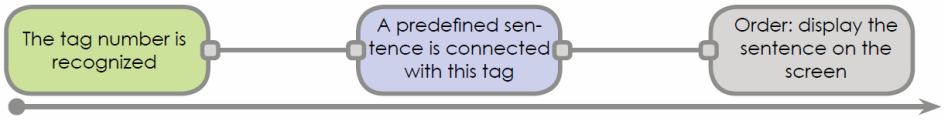
In your box, you have NFC tags. They will allow you to trigger actions just by passing them above TimeSquAir. Please take the one with the logo of TimeSquAir and pass it on your TimeSquAir: a message will be displayed on the LED matrix!

To learn more about these NFC tags, go to the "Trigger with NFC tags" tutorial.

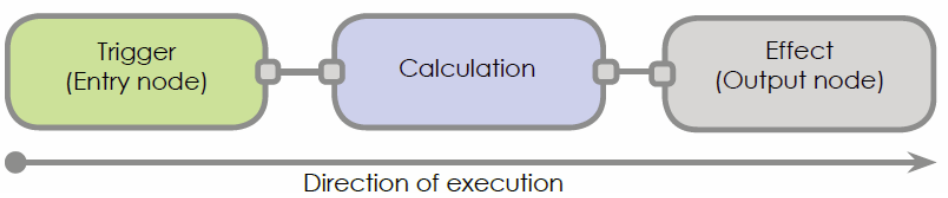
How it works

When you passed the tag on TimeSquAir, an action has been triggered : to display a sentence on the screen.

Here's what happened:



Which corresponds to the following scheme:



The three blocks that you see are the **nodes** : they compose your application.

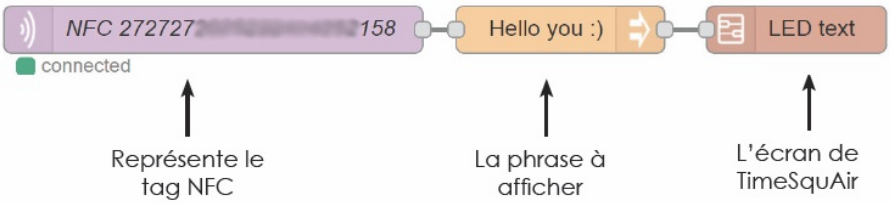
When nodes are interconnected, "wired", they form a chain of nodes called a **flow**.

TimeSquAir always needs a **trigger**: in this case it is the tag, but it can also be a tweet, a click on the page, an alert ...

Then you must tell him what to do with this trigger, define an action associated to the information received.

When the trigger occurs, it sends a message on the wire to the following node, from left to right, which uses it and then sends it to the following node, until the last node is reached.

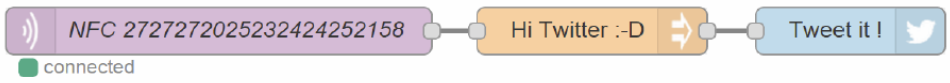
In the Visual Editor (see **The Visual Editor tutorial** to learn more about it), you have the following flow:



Below are some more examples of the trigger/action scheme:

Display the weather in Paris:

Tweet a sentence:



Every TimeSquAir application uses the same format: a trigger will cause an action (of which you define the settings). You choose the trigger nodes, nodes of calculation and output nodes.

The Trick

The trick used by the flow to integrate nodes is the wire. Every node knows the structure of the message on the wire. Thus, any node built can interract with any node.

Where to go now

- Use the tutorial "**The Visual Editor**" to discover IBM's Node-RED that allows to modify the behavior of your TimeSquAir.

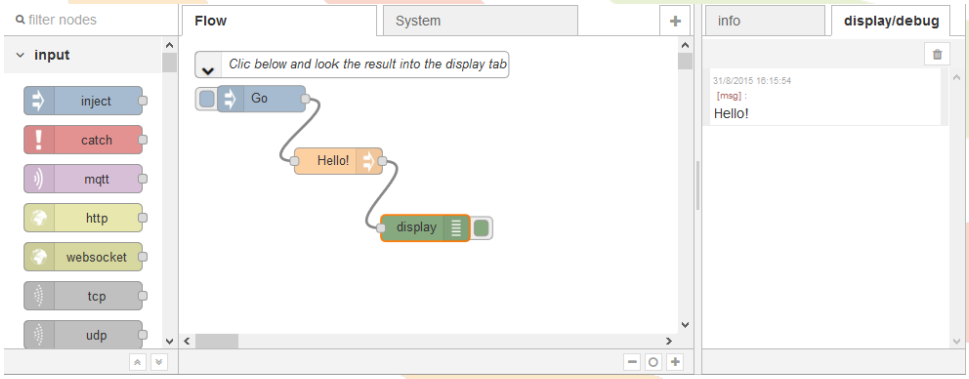
The visual editor

This tutorial introduces the visual editor: a web page (that you access from your internet browser) displaying a graphical interface that allows to create new applications and change preferences.

The visual editor is IBM's Node-Red (**http://node-red.org**) an open source visual editor for using the internet of things.

It is a very dynamic project with an ever growing community.

Discovering Node-RED



Palette Workspace Display / Info

The **Palette** contains all available **nodes**.

Nodes can be wired together to build **flows**.

In the flow above, clicking "Go" goes through "hello" and ends in the display node that prints "Hello" in the display tab.

How to build a flow

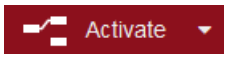
Building a flow is easy:

- Drag and drop nodes form the left palette
- Wire them together by drag'n dropping their handles

Activating

Once modified (by adding or removing nodes or wires) the flow should be activated by using the activate top right button.

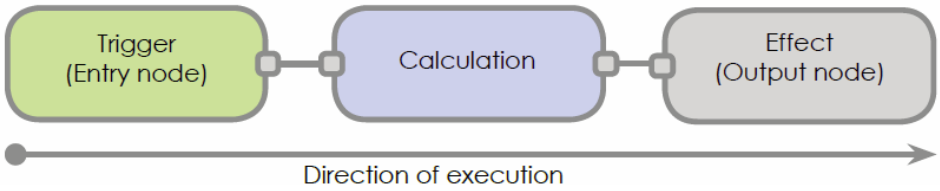
Activating pushes the flow from your desktop web page to TimeSquAir that takes it into account.



Trigger the flow

The flow is triggered by a manual event (as the "Go button" above) or by external events (tweets, NFC tag, ...). When triggered, the execution starts with the trigger node and goes right through the flow. If a Debug or Display node is invoked, a string is displayed in the info/debug tab.

As stated in the "**Try and understand**" tutorial, flows follow the scheme below:

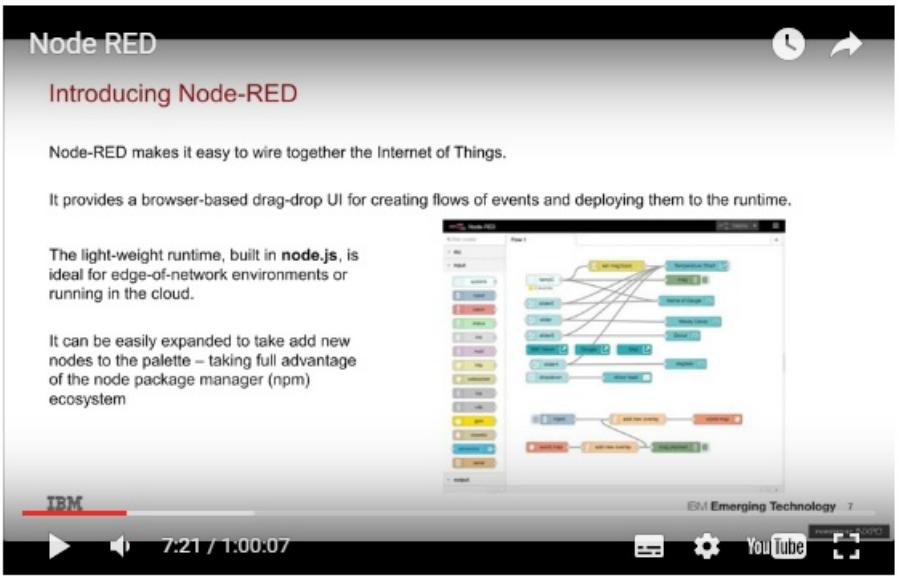


Learn more about Node-RED

- First read the IBM's Node-Red website (**http://node-red.org**).
- **noderedguide.com** series of lectures designed to get you started with Node-RED

You can also go to Youtube.com to get great video tutorials

- Node-RED innovators Dave Conway-Jones and Nick O'Leary provide an overview, demo, and technical details for their innovation that's making a big splash.



https://youtu.be/r3_L8zgYUkE

- Read (and contribute to) the forum **https://groups.google.com/forum/#!forum/node-red**

How to find new nodes

Browse the IBM's repository: **http://flows.nodered.org**. There is a great Node-RED community that publish new nodes every day.

Use the "**Import Node tutorial**" to learn how to import nodes.

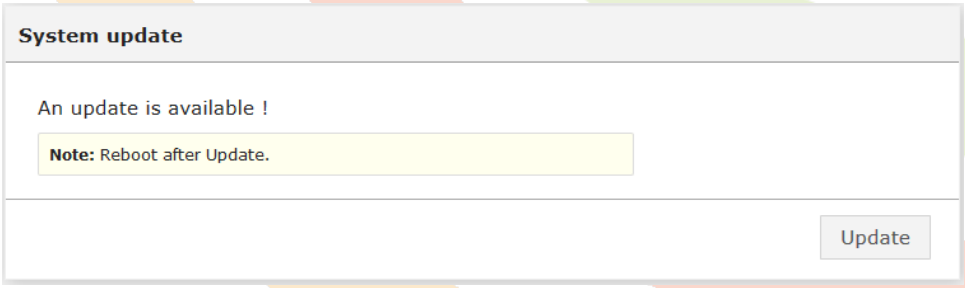
Update

Updates are frenquently published on the cloud to add new functionalities (ok, sometime also to fix bugs).

There is a button to automatically update to the new release.

Is there an update ?

Use the right menu and choose the "Settings option". At the bottom of the page you can get the current version, and before it, if an update is available or not.



Update!

If an update is available, clik update, wait (sometime a lot) and reboot.

Limitations

Unfortunatly, it is not always possible to update all the system, particularly if it involve low level code. Thus, the updated device may differ from the current release that can be downloaded from the web site.

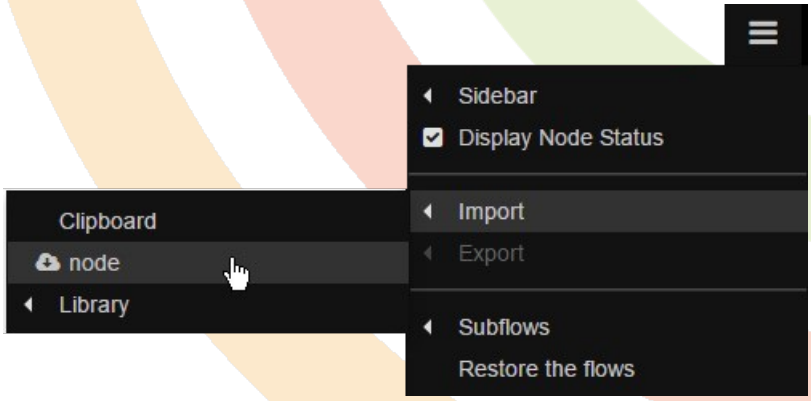
When possible, flash a new image.

Import Nodes menu

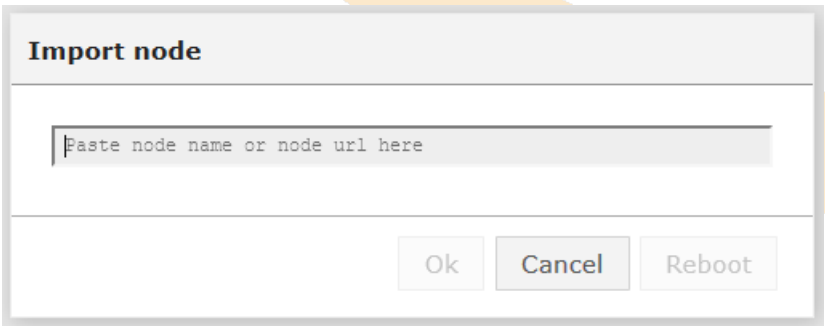
There is a big community that builds software for Node-RED and the Pi. The import Nodes functionality is a dialog that allows to add third party software to the current installation.

Adding new functionalities

Use the following menu:



And fill the edit field with the libray name to import



- The libraries are published here: **http://flows.nodered.org**.
- The libraries dedicated to the Thingbox can be found here: **https://www.npmjs.com/search?q=ttb-**

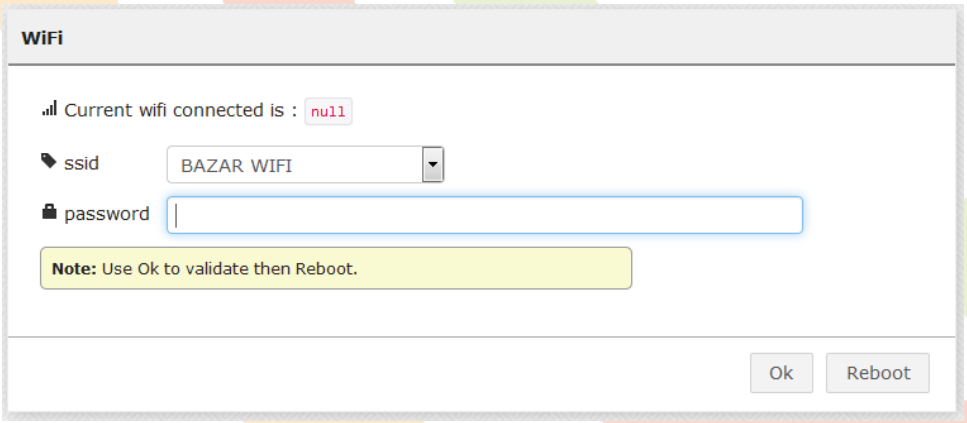
Be aware that some nodes need further low level installation (as "apt-get xxx") that can be only be made by advanced users. These additional steps cannot be guessed by the generic node installer that is described here.

Connecting with WiFi

WiFi allows to connect wirelessly to internet.

Edit the WiFi Settings

From the Settings option of the right menu, go to the settings page and look at the WiFi setting box:



Enter the name of the WiFi network and the password. Hit Ok and then reboot. It should be ok then.

Old devices

On old devices before Pi 3, you may need a dongle in the USB port. In this case, choose the following dongle **"Edimax EW-7811UN Nano Adaptator"**.

Using the LED Matrix

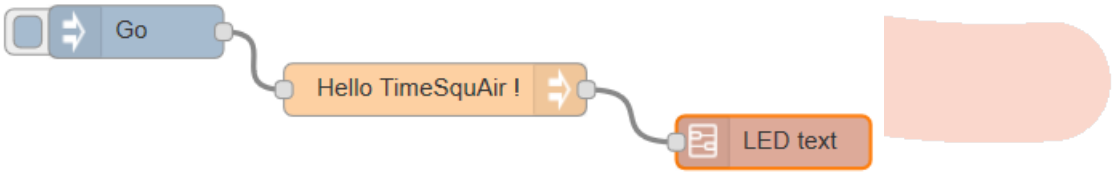
The LED Matrix is a 16 x 32 LED display that stands in front of TimeSquAir.

This tutorial shows how to use the provided nodes to use that display.

The LED Matrix "subflow"

A **Subflow** is a node made with a flow. There is two ways to make a node: one way is to use the javascript programming language, the second way is to use a flow.

To easily address the most frequent use, there is a orange node that does the job of displaying on the matrix:

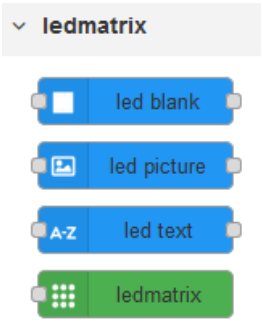


You can use this node to display simple texts.

To do more, as mixing text and pictures, use the nodes below.

The LED Matrix nodes

You can find the LED Matrix nodes in the palette:

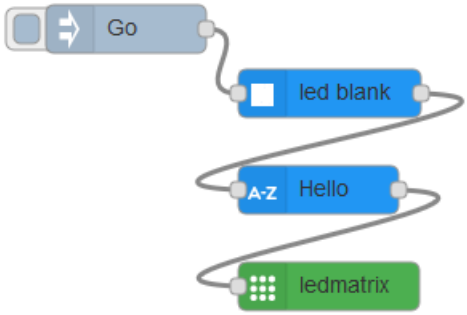


First use a LED Blank node to start. Then use a mix of LED Text and LED Picture nodes to compose the message. End with a LED

Matrix node and use a Go node to activate the whole.

Display a simple text

To display a simple text, a LED Text node is enough:



Double click the LED text node to edit the text to be shown.

See later if the text to display should be calculated.

Display a picture

Add the picture file on TimeSquAir

In order to add a picture on the LED matrix, you have to copy the picture on TimeSquAir.

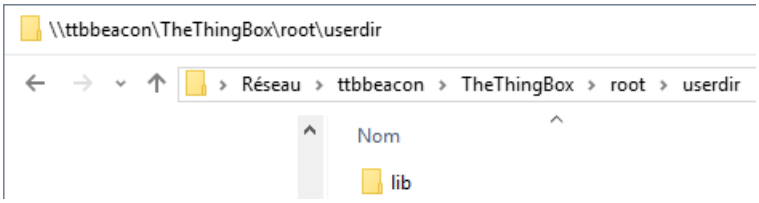
Just get the destination folder using Samba: in your PC file browser, open:

```
\\myTimeSquAir\TheThingBox\root\userdir
```

where "myTimeSquAir" should be replaced by the name of your TimeSquAir.

Windows 10 asks you for credentials? Follow the following tutorial to fix your network: **https://tinkertry.com/how-to-change-windows-10-network-type-from-public-to-private**

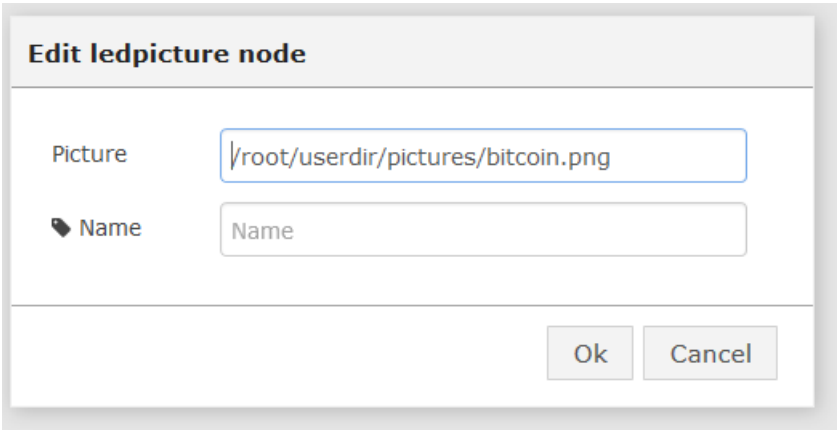
You can do it by opening a file explorer window and changing its address by this one in the address bar. In the picture below, I access my TimeSquAir which name is "ttbbeacon" like that:



In you userdir, find or create a folder named `pictures` and put your pictures in it. You can use `png`, `jpg` or `ppm` images.

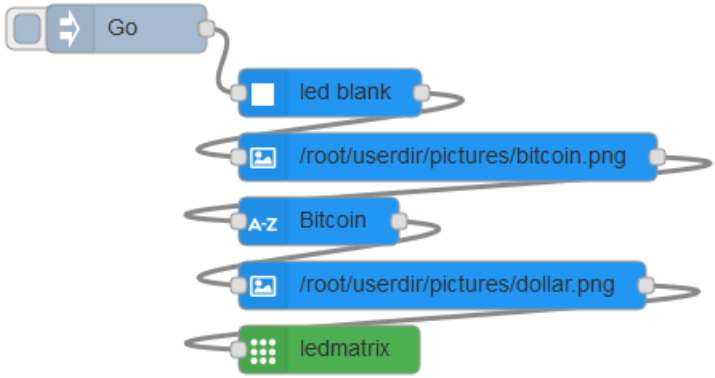
Modify the flow

Add a LEDPicture node and fill it with the path of the picture:



and insert it in the flow.

Below is an example of mixed text and pictures:



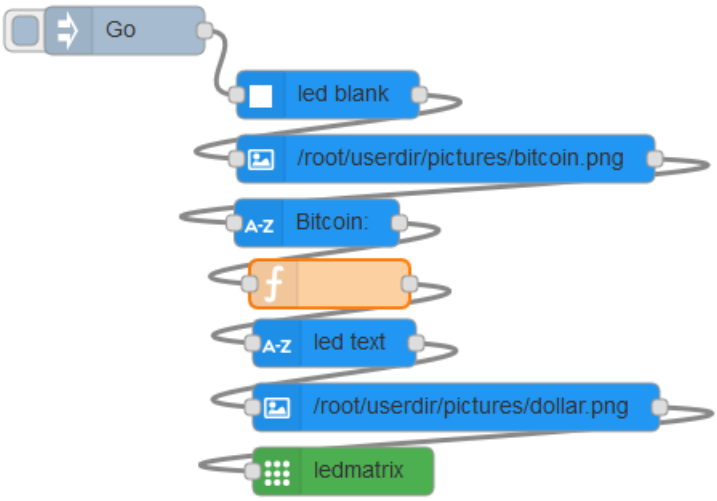
Using calculated values

Of course, some value may be unknown at this time, thus should be calculated at runtime.

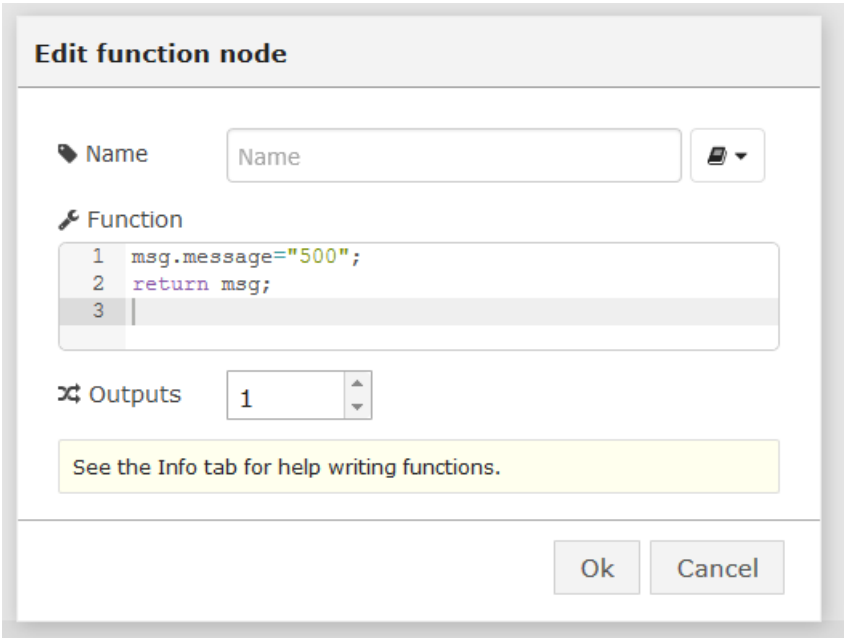
The solution is to use a function node and calculate the value in javascript (an easy software language).

A basic example

Here is the flow with a new function node that sets the message to "500":



Double click on the function node to add the javascript:



- **line 1**: the `message` property of the flow (the msg object) is set to "500". We do that because we know (its written in the info tab) that the LEDText node uses that property to find the text to display. You can do any calculation using javascript to calculate the accurate message.
- **line 2**: we return the modified msg object to let it follow the rest of the flow.

Get a value from the cloud

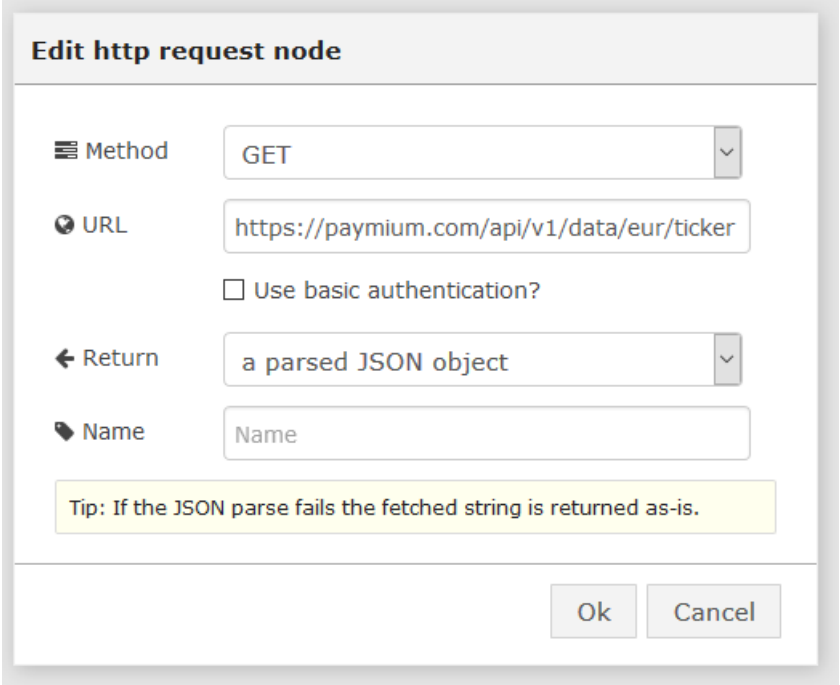
Make it more useful and go fetch the bitcoin value on the web.

We get the value by making an HTTP request to **https://paymium.com/api/v1/data/eur/ticker** which returns data

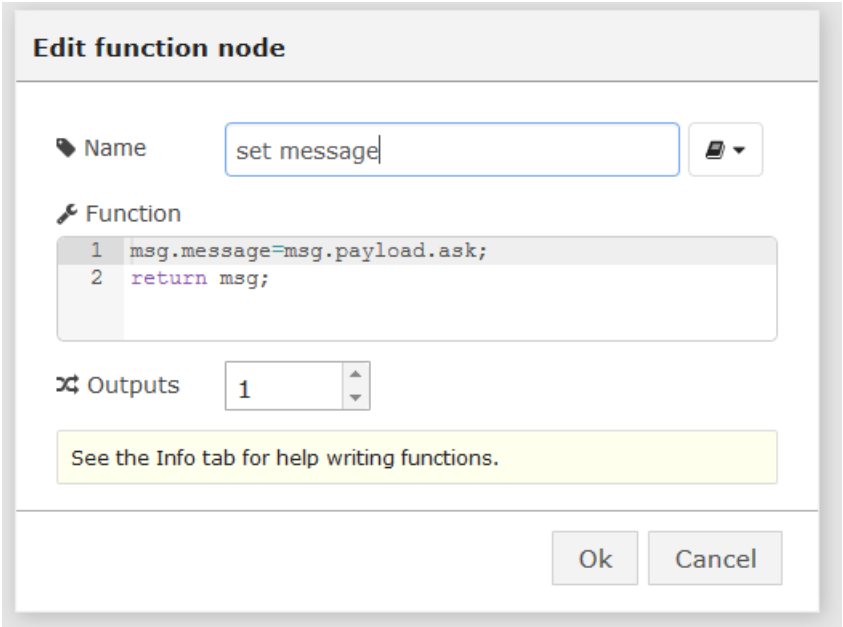
like:


```
{
  high: 374.99,
  low: 365.63,
  volume: 43.36850015,
  bid: 367.01,
  ask: 369.07,
  midpoint: 368.04,
  vwap: 368.91341449,
  at: 1459516121,
  price: 367,
  open: 365.62,
  variation: 0.3774,
  currency: "EUR",
  trade_id: "1f7d36f2-b66e-46a4-8c33-8568f0d7f8e0",
  size: 0.682
}
```

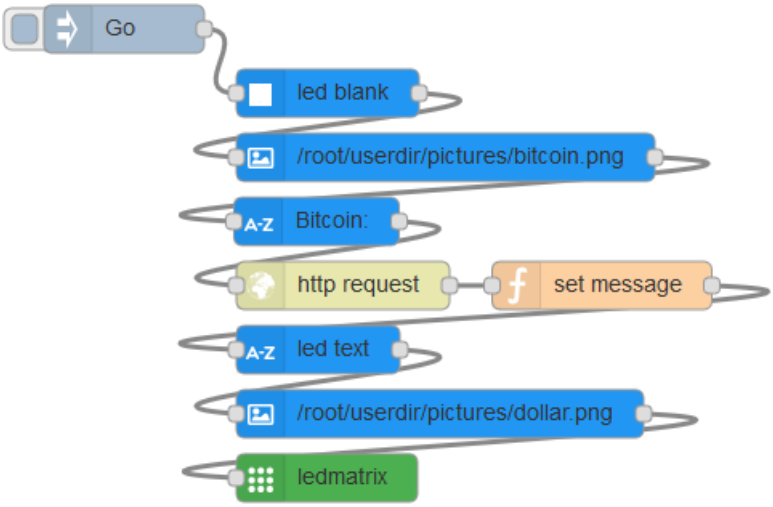

To do that, we use an "HTTP Request" node and fill it with the address above:



Then we link a function node to feed the msg.message as above:

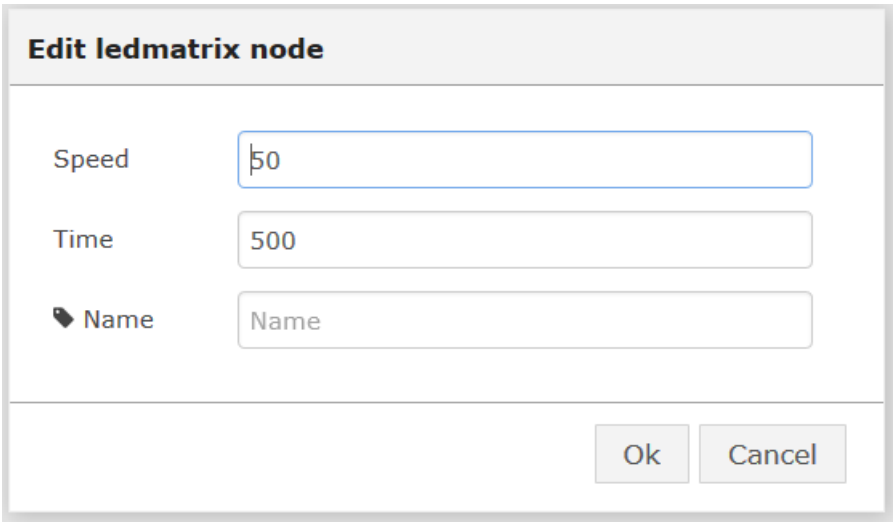


Here is the final flow:



Change display settings

By editing the LEDMatrix node, you can change the *speed* and the *intensity* of the display.



Trigger with NFC tags

Near field communication (NFC) is a set of communication protocols that enable two electronic devices, one of which is usually a portable device such as a smartphone, to establish communication by bringing them within 4 cm (2 in) of each other.

Learn more on Wikipedia (**https://en.wikipedia.org/wiki/Near_field_communication**).

The TimeSquAir box contains some NFC tags and the TimeSquAir contains a Tag reader at its top.

What is in NFC tags

The important thing to know is that a tag may contain basically two important parts:

- Always: One indentity number (ID) which is unique.
- Sometime: a memory part that can contain some additional information

Most of the tags you can get from hotels, clubs, underground access cards... are cheap tags that contains only an ID.

The trick is to use these unique IDs as a trigger of actions, so you can use all of these tags you get for free.

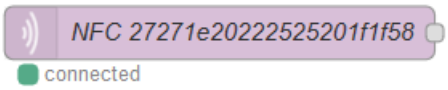
What to do with NFC tags

Here are some ideas:

- Your kid comes from school: it puts the tag on top of TimeSquAir and you get an email
- You have a restaurant. Customers gets pictures (with a NFC tag at the back) of what they want to eat. You build the order just by sliding the meat pictures on top of TimeSquAir.
- You have music CD covers. You can launch the music by sliding the cover on top of TimeSquAir.

Using NFC tags

Take a tag (except the branded one that is already included), slide it on the top of TimeSquAir and a new node will appear on the workspace:



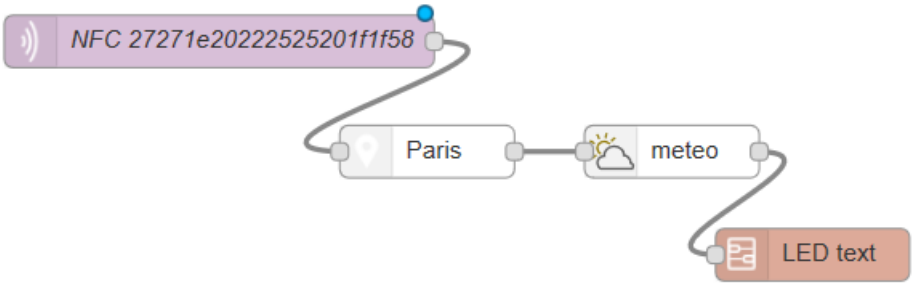
The big number is the unique identifier of the tag. You can change the appearance by editing the name of the node (double click it).

This will work only once! We already did it with the tag with the TimeSquAir logo so the node won't appear again with this tag.

This node is your tag: every time you slide, a message will go out of

it.

Let's use it as a trigger (this node can replace any Go or Inject node) to get the weather:



Send an email

There is a dedicated node to send an email.

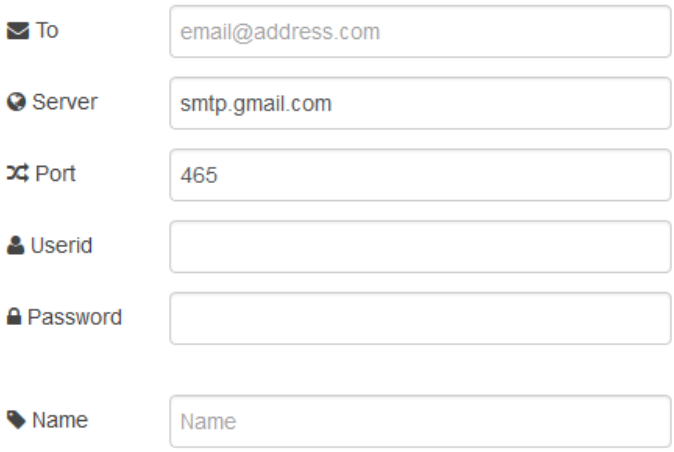
It uses an email server (named "SMTP") and you have to get one (a free good one in google mail (gmail)) and enter its parameters into the node.

Caveat: it seems it is not allowed to use Hotmail as a server to send emails.

E-mail node

Drag and drop an email node to send the message:

And parametrize it to send the message:



- "To" : fill with the recipient
- "Server" et "Port" are the server parameters (for gmail, use "smtp.gmail.com" and "445")
- "Userid" is the user (and "Password" its password) allowed to use the mail server above (for gmail, use your account xxx@gmail.com).

GMail

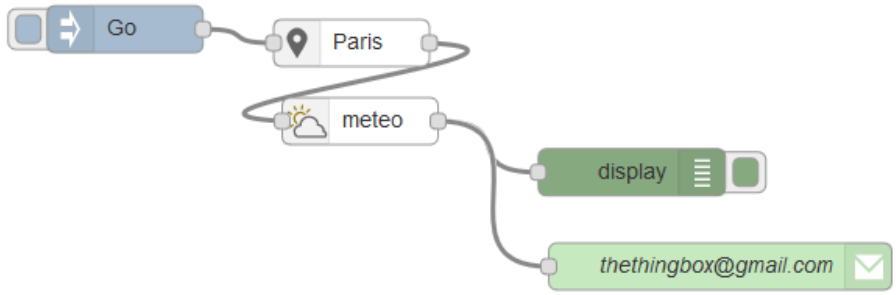
If you use a Gmail account to send the email, you have to check a box in the Google settings:
https://www.google.com/settings/security/lesssecureapps

More info about gmail servers settings
http://www.hesk.com/knowledgebase/?article=72

http://email.about.com/od/accessinggmail/f/Gmail_IMAP_Settings.htm

Build a flow

Let's use the weather as an example:



msg :
Paris:3°C

Some music !

This tutorial will show you how to use the node Sound to play some music radio from the web.

Multiple sounds can be played at the same time (a beep or an alert on top of a music) by using two Sound nodes.

A minimal sample

The blue node is the sound node from the palette:

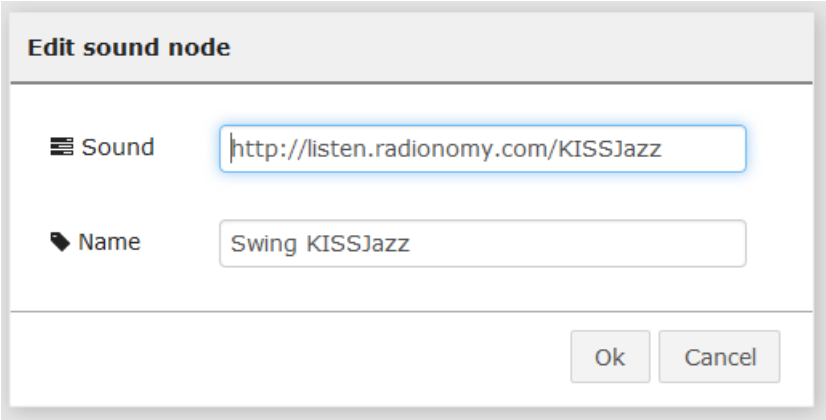
Drag it to the workspace and double click it.

You can choose any track on the web or on your Pi if the path is correct.

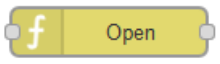
However, be aware that your file must not have any space in it's name !

To take an example, we will use this radio available on the web :

http://listen.radionomy.com/KISSJazz

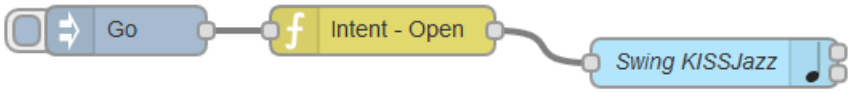


Now you need to start the music. There an "intent" for that: a node that puts in the flow intentions as "open".



Drag a Go node in front of it to launch the flow.

Here is the result:



Clic on the red button "activate" on the upper right corner to activate the flow changes.

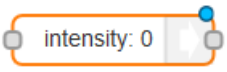
Then launch the flow using the Go node. If a HiFi device is plugged into the jack of the Pi, you can now ear music!

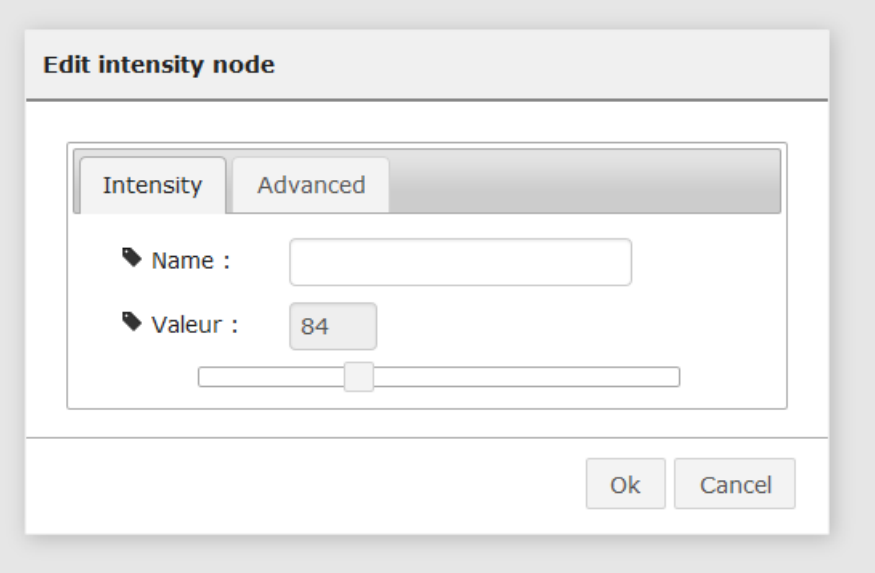
Stop the music

Now you are also able to add the "close" intent node and use it to stop the music.

Change the volume

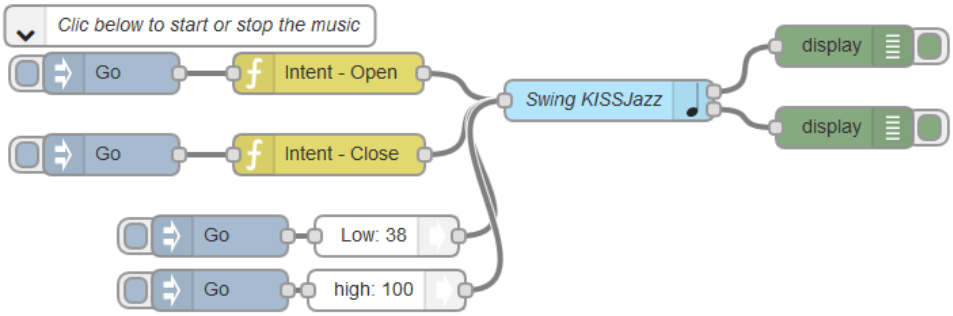
To change the volume, use the intensity node intent. It is use to set the level of something.





You can use the value 100 pour high level, and 40 for moderate (as you feel).

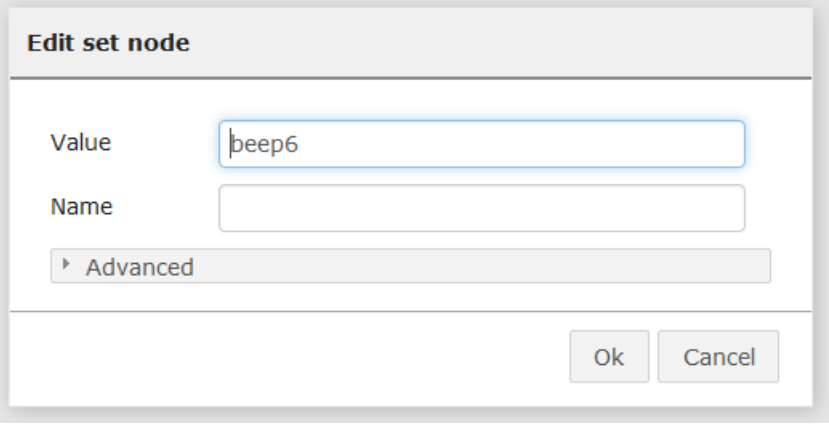
Link them as below:



I added two display nodes so I can see any error message in the display tab.

Play a beep

There is 50 predefined sounds you can play with the Sound node:



A full path (beginning with /) can also be used, with an extension if non mp3 file.

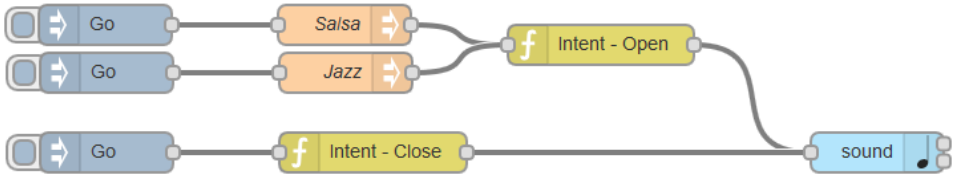
More Radios !

There is a library (use the menu on the right, Import/library/Sound Demos) with some radios you can play with.

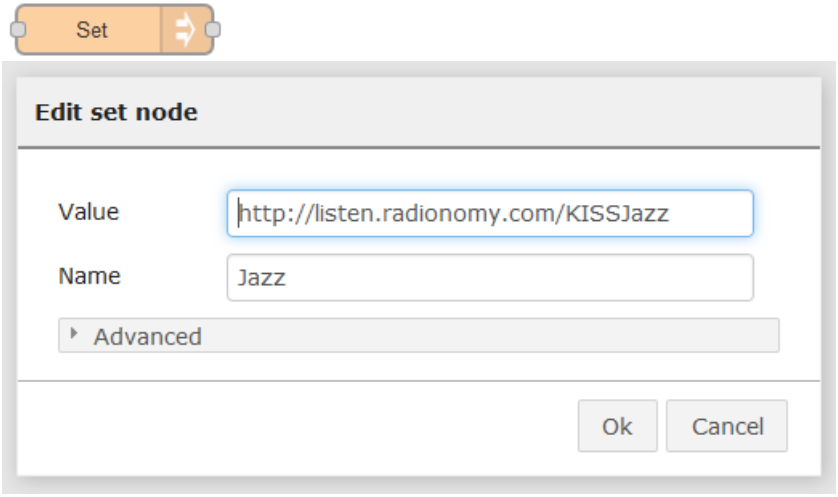
Want to go further ?

In this tutorial, we set the music by editing the node.

It is possible to "calculate" the music adress before entering the sound node, so it can depend of the context (salsa when its cold, Jazz when its hot).



It uses the Set node that changes the value of a particular track of the flow.



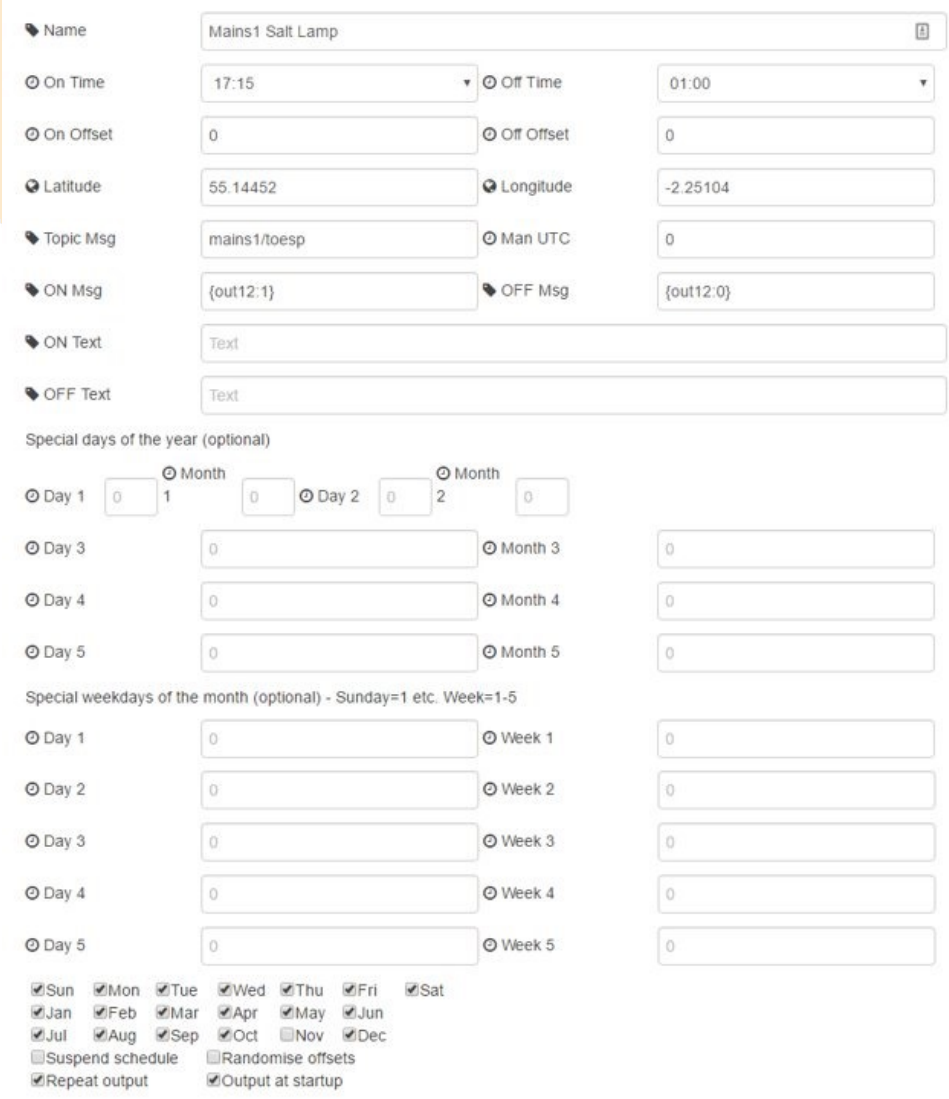
Now you have to replace the Go nodes by context triggers...

Credits

Based on the wonderful lib **https://github.com/turingou/player**.

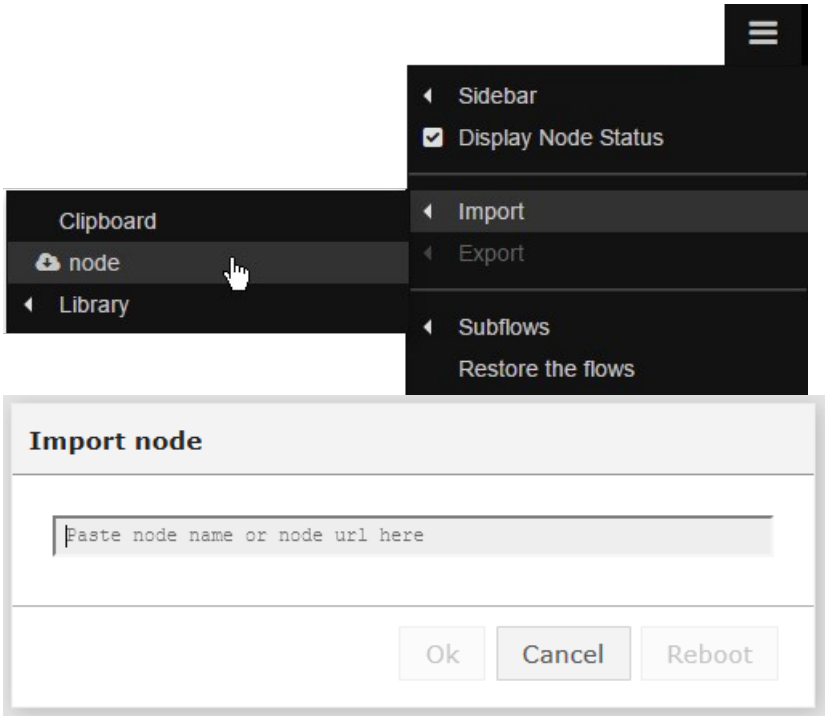
A scheduler

Use this scheduler to trigger event at a particular date, "at dusk", "at dawn", " dawn every day – but not in December through February" ...



Install the node

Click the right menu in the Node-RED web page. Choose `Import` then `node`.



Type

`node-red-contrib-bigtimer`

in the edit box and click the OK button.

This imports a third party software. The installation may take minutes (a progress bar shows the progress), depending on what is already installed on your device.

At the end, click the `reboot` button to restart the device.

Now you can find this node in the palette (use the filter to find it):

Drag'n drop it on the workspace, and double click it to set parameters:

If you accept to share the location in your browser, the fields will be filled:

Trigger the flow with the scheduler

BigTimer triggers the flow when scheduling conditions are met.

They can be set by editing the node (double click it).

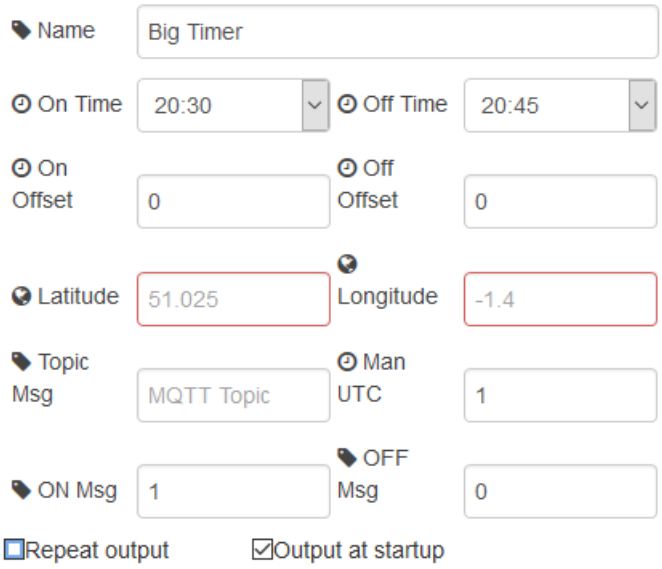
Learn more about bigTimer settings:

- **http://tech.scargill.net/big-timer/**
- **https://www.npmjs.com/package/node-red-contrib-bigtimer**

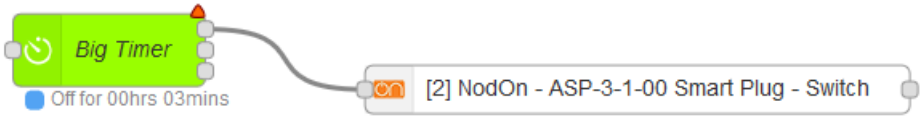
The following sample shows how to open a ZWave switch between two hours:

Fill:

- On Time
- Off Time
- 1 as onMsg
- 0 as offMsg
- remove repeat output check



Make a flow to monitor a ZWave switch (more on the switch in the **Nodon SmartPlug** chapter).



Ressources

- Learn more about bigTimer settings **http://tech.scargill.net/big-timer/**
- Learn more about ZWave integration in the **ZWave chapter**
- more on the switch in the **Nodon SmartPlug** chapter

Credits

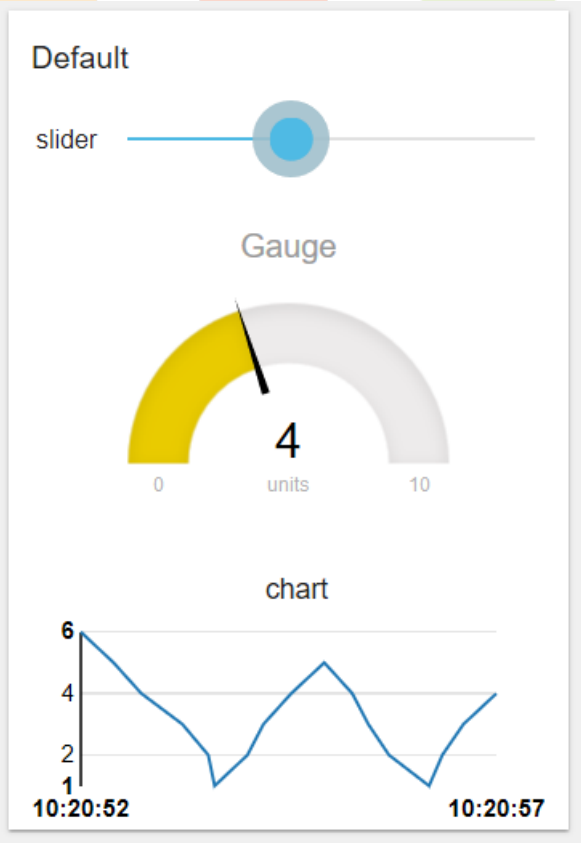
Big-Timer is created by **Peter Scargill**

An easy Dashboard

Create a web dashboard from your Node-RED Flow running locally (no cloud).

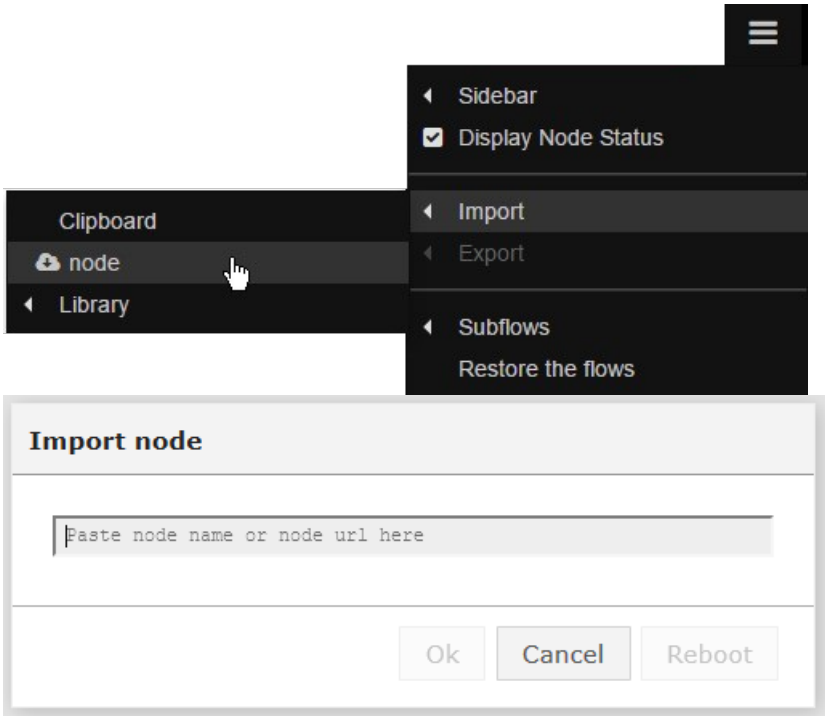
This tutorial shows how to insert web widgets into the NodeRED flow and use them from a web page where they are gathered.

Here how the result looks like:



Install the nodes

Click the right menu in the Node-RED web page. Click `Import > node`.



Type

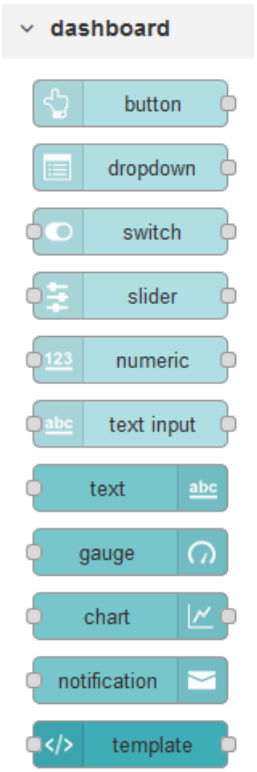
`node-red-dashboard`

in the edit box and click the OK button.

This imports a third party software. The installation may take minutes (a progress bar shows the progress), depending on what is already installed on your device.

At the end, click the `reboot` button to restart the device.

Browse the palette and look at the newly added nodes:

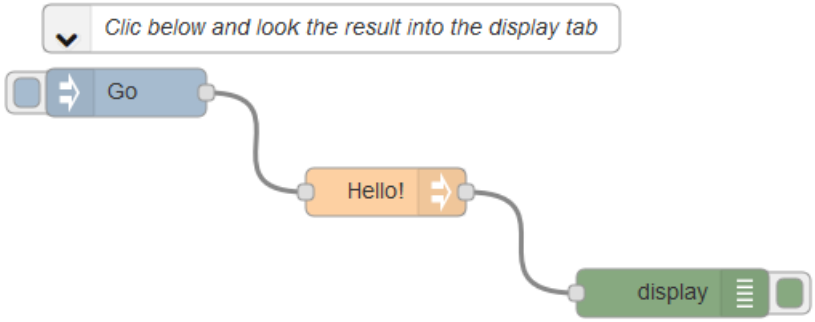


Refresh the browser if the nodes are not there

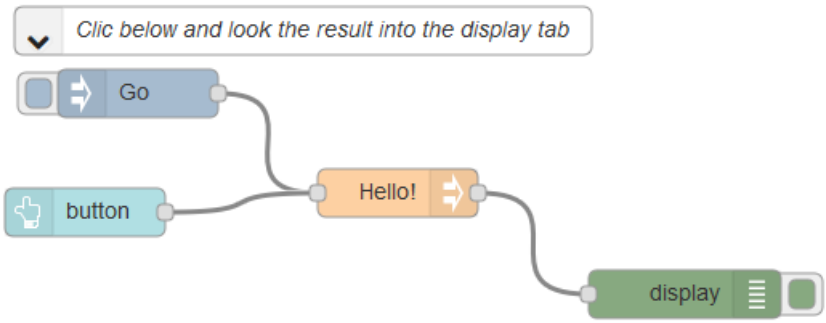
A first simple flow with a button

Add a Web widget node

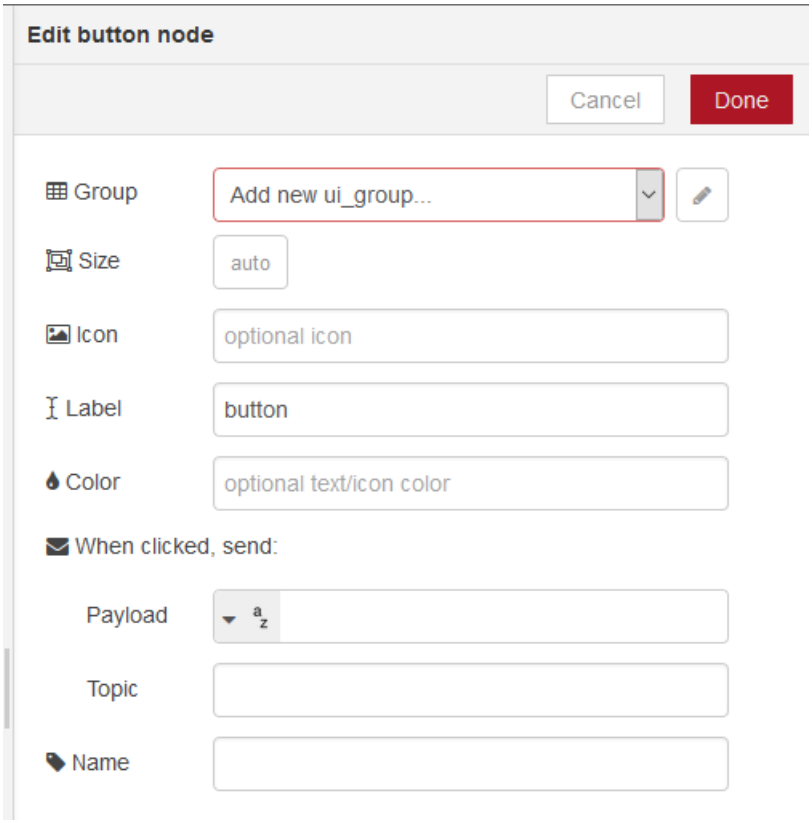
Let's start with a simple flow:



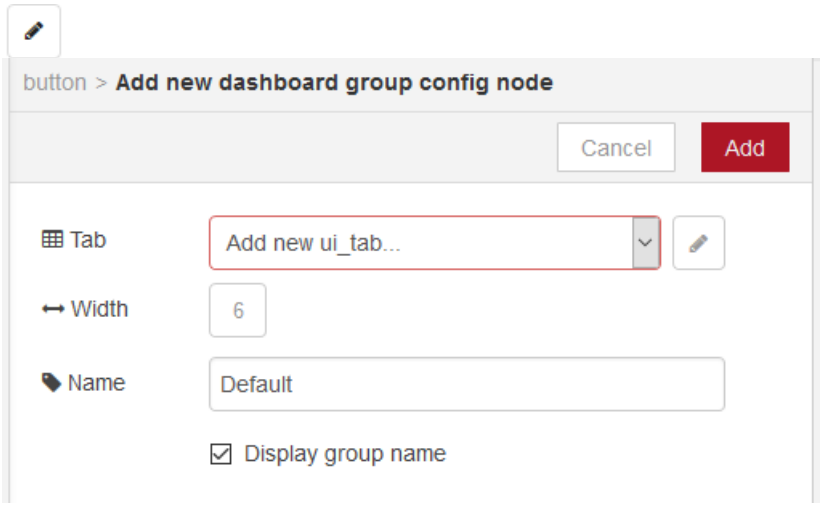
Drag the `button` node to near the Go one like this:



Double clic on the button node to add it in a UI group:



And use the following to create the UI group:



And use again the following to create a tab for the UI group:

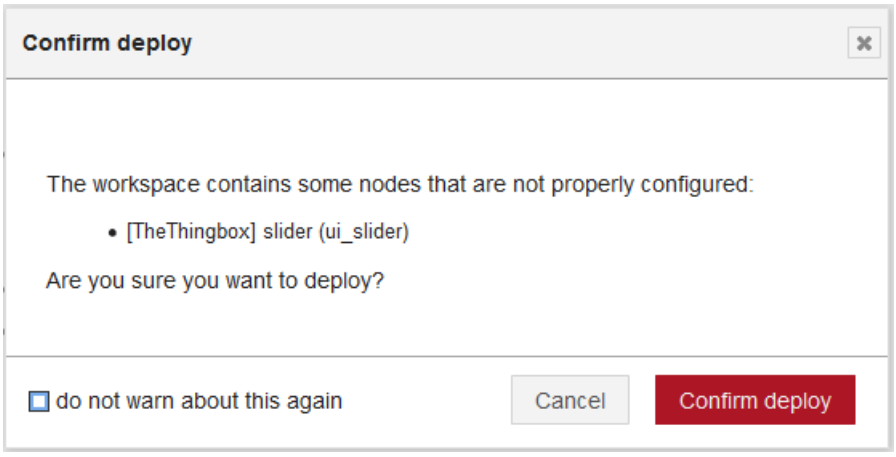
Just choose `add` without changing anything.

Then `add` again then close the previous dialog with `done` button.

Don't forget to validate the flow change with the `activate` button.

"Not properly configured" error

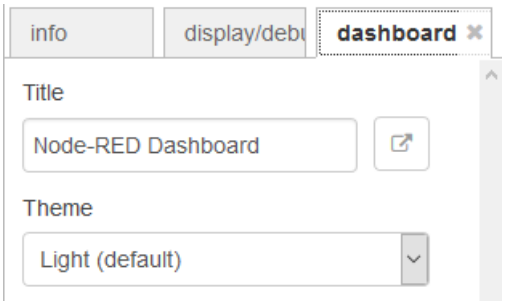
If you come to the following error below, that juste because node have no default settings when dragged.



Just double click the node - that adds the default configuration - and click the Done button. That's all.

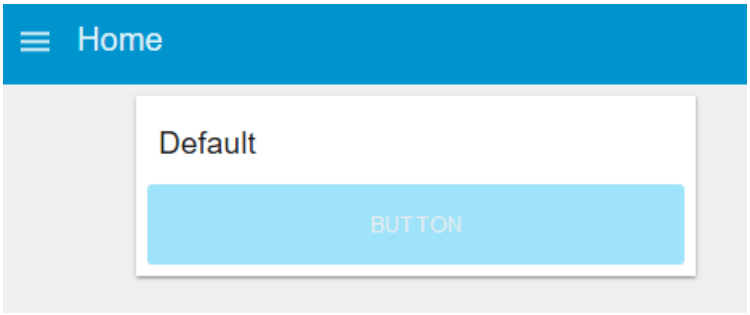
And see the result!!

Select the config tab on the right:



and open the dashboard by clicking the following icon:

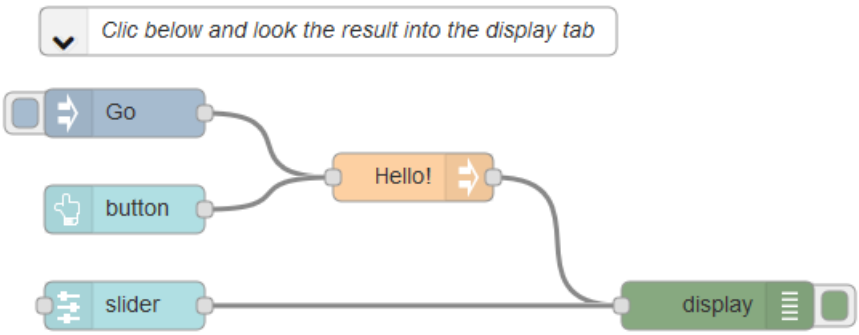
Here is the result:



You can push the button, it behaves as you push the Go button in the flow!

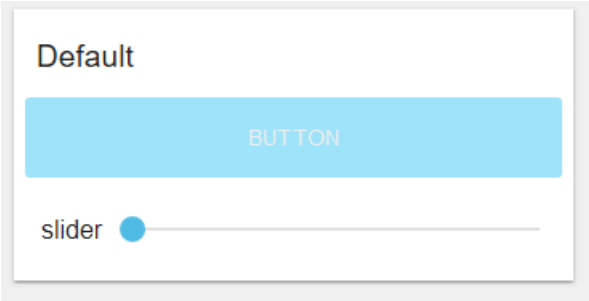
A simple flow with a slider

Drag the `slider` node like this:

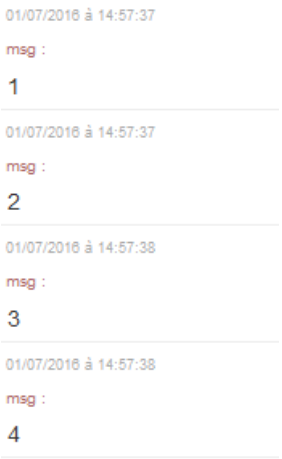


Double click the node to add a group as above.

The UI web screen becomes:

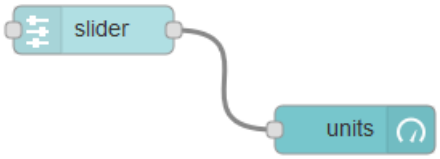


You can drag the slider: the output is according to the slider position:



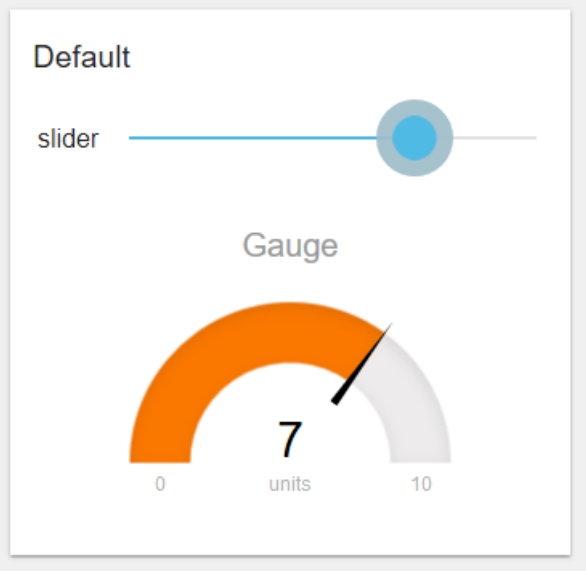
A simple flow with a gauge

Drag the `gauge` node like this:



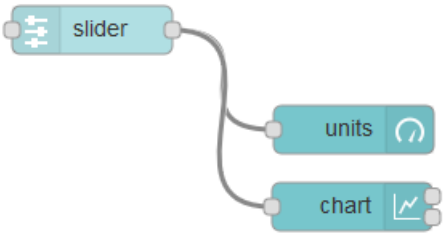
For a unknown reason, the name of the node becomes `unit` on the workspace

Moving the slider updates the gauge:

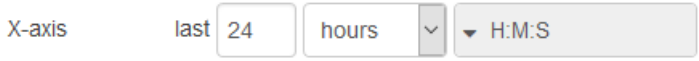


A simple flow with a chart

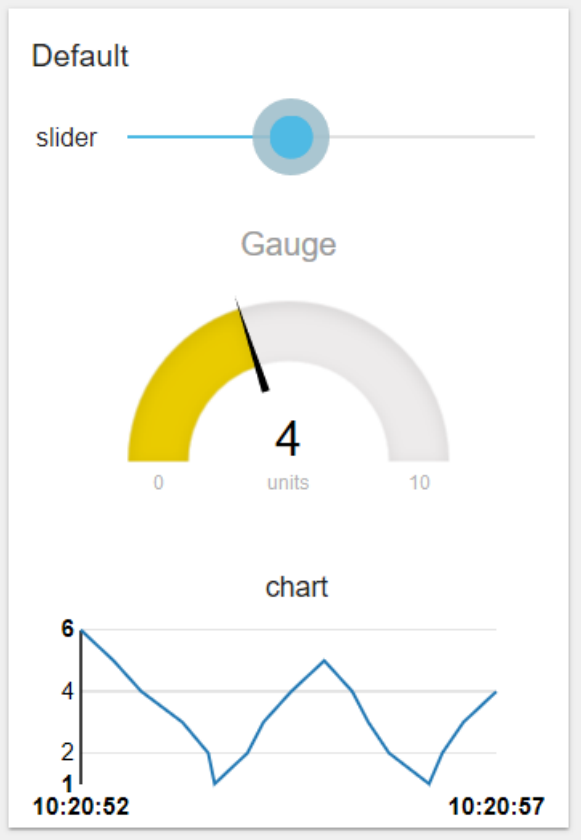
Drag the `chart` node like this:



Be aware to set an appropriate time window:



Moving the slider updates the gauge and the chart:



Use it with an Enocean temperature sensor

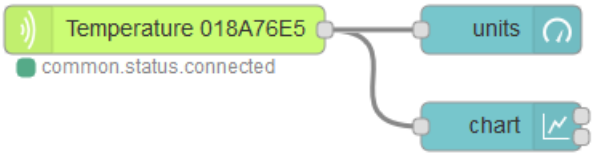
Install Enocean stuff as described **in the Enocean documentation**.

You also need a dedicated dongle:

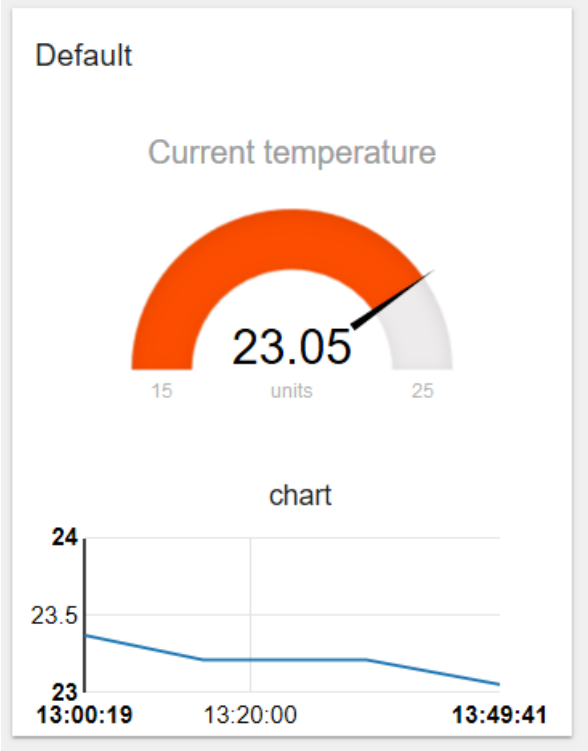
But then you can include a temperature sensor by pressing its LEARN button (learn more about this in the dedicated documentation "**Temperature sensor**"):

Nodon temperature sensor.

A node will pop on the workspace. Use it to build the following flow:



And get the following charts:



Using iBeacons

iBeacons are bluetooth devices that broadcast their information (position...) so that bluetooth aware devices (like smartphones, Raspberry Pi...) are able to detect them and get their information.

Learn more on Wikipedia (**https://en.wikipedia.org/wiki/IBeacon**).

In this tutorial, we show how to get the distance of an Estimote beacon (**http://estimote.com**) from the Pi.

Add a Bluetooth 4.0 dongle

You need to have a Bluetooth 4.0 dongle (you can buy it here **pimoroni.com** or **adafru.it**) and it should be plugged in a USB port of the TimeSquAir

Remember that only Pi3 has a bluetooth feature! add a dongle for previous Pi releases.

What to do with beacons

Some ideas:

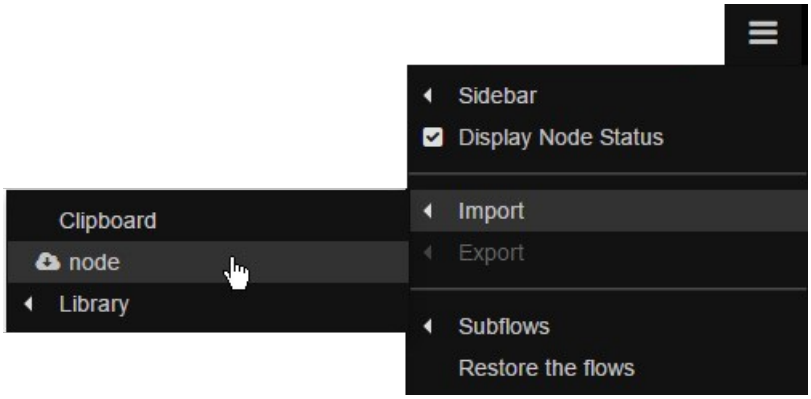
- Put a beacon in the bag of your kid and get an email when he comes back from school
- Trigger some security when you go far your desk
- Add a beacon to an important key and lights up a red light while the key is not back

Install Beacon software

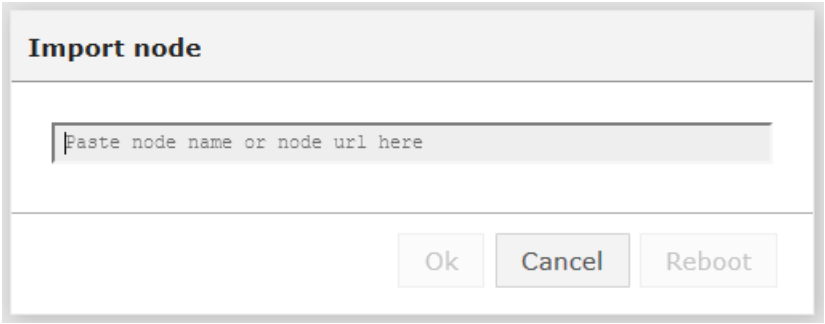
Some software is needed in Node-RED to communicate with the Pi.

To install it :

- Use the menu `import / node`



- type `ttb-ibeacon` in the dialog box.

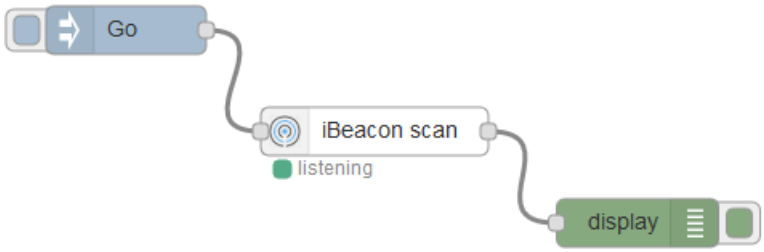


- Wait for the install
- Then reboot and then refresh the web browser page.

Check the palette (using the filter). The iBeacon node should be present:

Scan beacons around

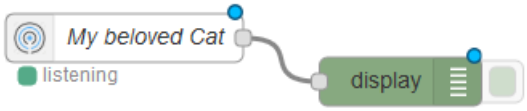
Build that simple flow. It will scan around and create a node for each found beacon:



When you click "Go", one node per beacon will be added to the workspace. The big number on it is some identifier.

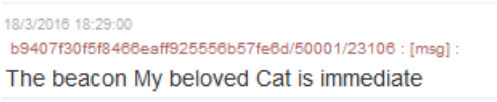
Use a beacon node

Double click it to change its name to something better (we suppose here that the beacon is wrapped to the cat), then add a Go and Display node:



Do not forget to activate (with the button with the same name). The blue bullets will then disappear.

Then the display tab gets filled with



There is one line each time the beacon sends a position information (quite often).

The message on the flow contains more information: use a debug node instead of the display to see the distance in meter into the payload:

The debug node also allows to see the complete msg:

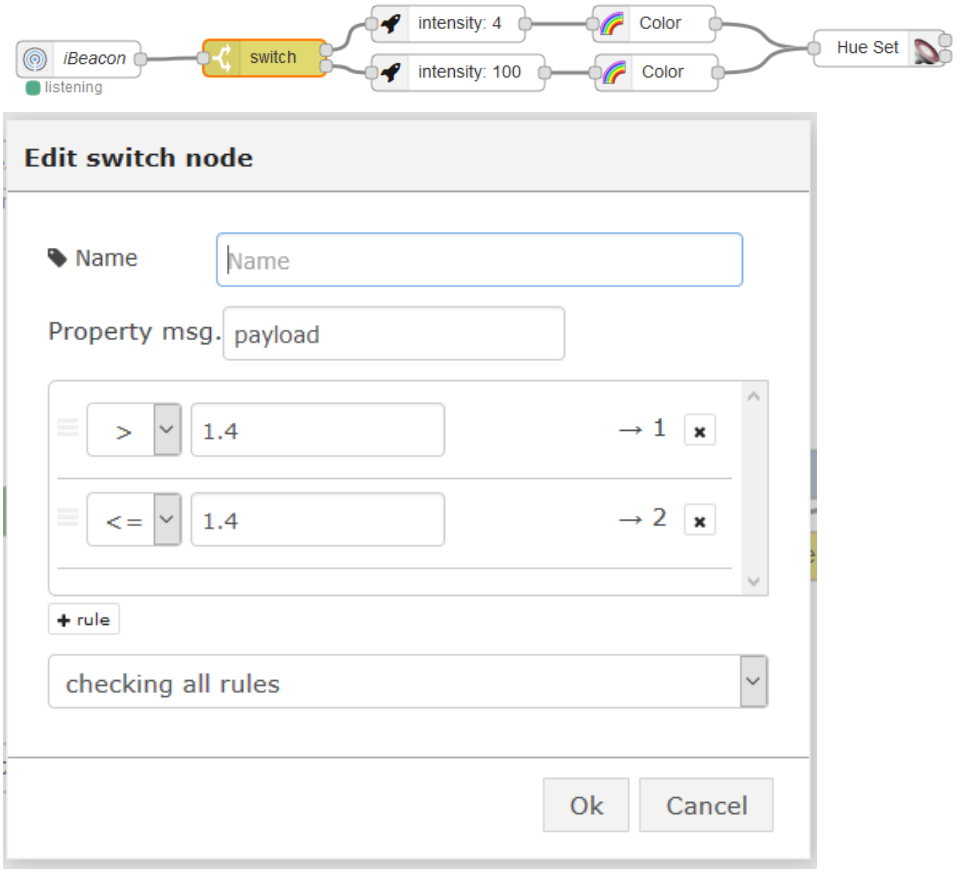
20/3/2016 23:39:59 [fbab7cb1.56e48]
b9407f30f5f8466eaff925556b57fe6d/50001/23106 : [msg] : object
{ "uuid": "b9407f30f5f8466eaff925556b57fe6d", "major": 50001, "minor": 23106, "measuredPower": -74, "rssi": -87, "accuracy": 1.924770003766445, "proximity": "near", "topic": "b9407f30f5f8466eaff925556b57fe6d/50001/23106", "payload": 1.92, "message": "The beacon 'iBeacon' is near", "_msgid": "e357b24e.1ca85" }

Do not use "accuracy" property which seems inapropriate

Trigger events

The distance value in the payload can be used to trigger events.

Use a switch node and add a conditionabout the distance of the beacon:



and use it to trigger a light when the cat comes near its food!

Google Physical Web

Publish "hot deals" through the Google Physical Web

The Physical Web, a project from Google's Chrome team, is an open source approach to allow physical devices (e.g. a vending machine, a poster, a toy, a bus stop, a rental car) to broadcast URL around providing an access to information or remote control.

Google provides phone Apps that allow the user to discover the brocasted URL and use them to access web pages, javascript apps or nativ apps to get information about the device or remote control them.

Get more info about the physical web here:
http://google.github.io/physical-web/

- Download the Physical web Android app here **https://play.google.com/store/apps/details?id=physical_web.org.physicalweb**
- Download the Physical web from the Apple Store here **https://itunes.apple.com/fr/app/physical-web/id927653608?mt=8**

More on this Google video:

http://www.youtube.com/watch?v=1yaLPRgtlR0

This tutorial describes steps to publish a store "hot deals" as a

picture on the Physical Web. Without any technical knowledge, you'll be able to use a Raspberry Pi to brodcast URLs around that can be seen with the nativ apps.

We'll use two steps:

- publish a picture on the Internet with **postimage.org**
- broadcast that picture URL with the Pi using the Physical Web bluetooth protocol.

After that, any customer using the Physical web app on his phone will have access at the URL and then the picture, displaying the "Hot Deals" of your store.

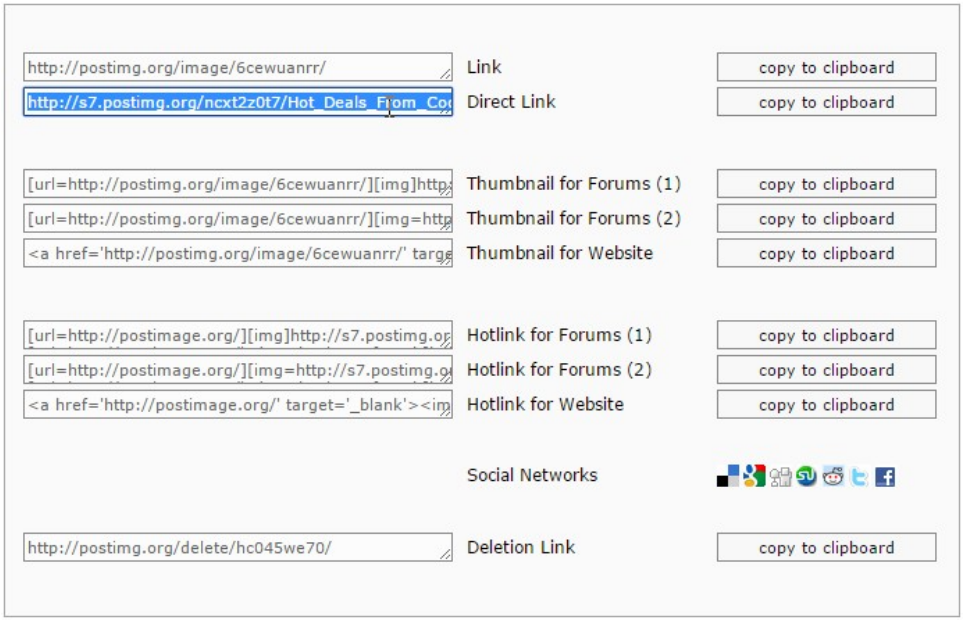
Except for Pi3, you need a Bluetooth 4.0 dongle (you can buy it here **http://adafru.it/1327**) and it should be plugged in a USB port of the TimeSquAir

Publish a "hot deals" picture on the internet

Your hot deals web page must be available through internet. There are several solutions and the chosen one here is **postimage.org**:

Upload your hot deals image on postimage.

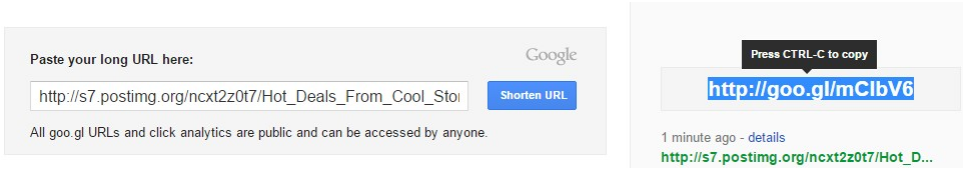
Copy the `Direct Link` of your hot deals. Also copy the given URLs to reuse them later if needed (as the deletion link).



Shorten URL

Because the length of the broadcasted URL should be short, we have to use an URL shortener.

Let's use the **Google shortener http://goo.gl**.



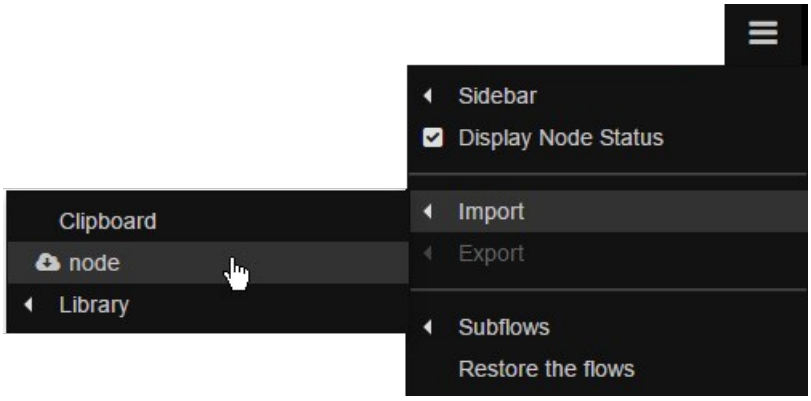
Copy your long hot deals URL and note the shortened version.

Access the visual editor from your browser

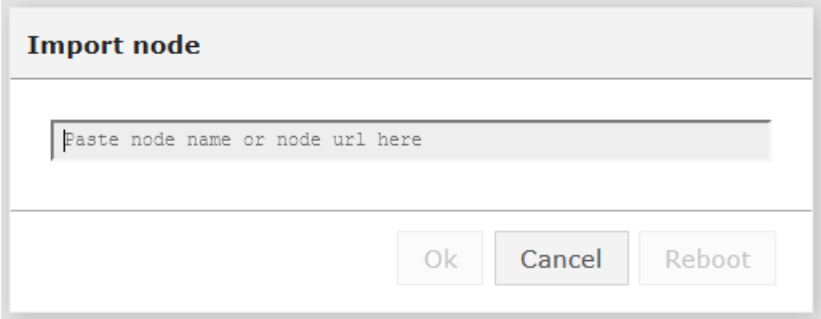
Enter the name address of the Pi into the address bar of your browser (see **here** for details).

Install the Eddystone node

Click the right menu in the Node-RED web page. Click `Import > node`.



Type `node-red-contrib-eddystone` in the edit box and click the OK button:



This imports a third party software. The installation may take up to 5 minutes (a progress bar shows the progress), depending on what is already installed on your Pi.

At the end, click the `reboot` button to restart the Raspberry Pi.

Step 8 : Build a flow.

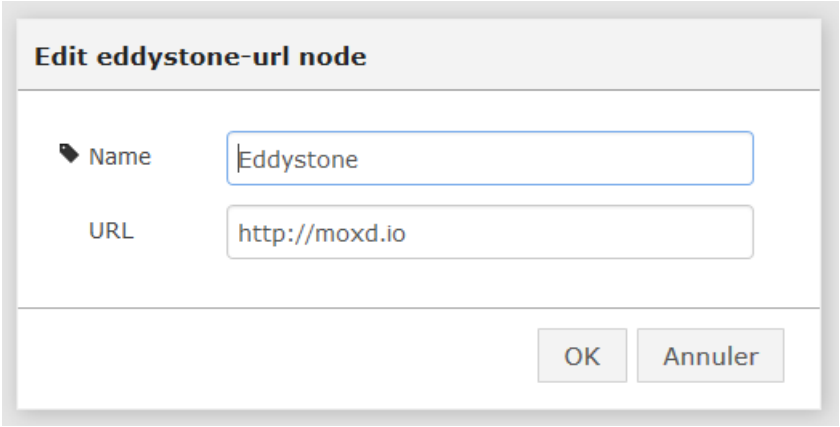
Use the menu `Import / lib / PhysicalWeb demo 1` to add an already available flow

Here is the flow:

"Eddystone" is the name of the protocol used to broadcast the URL with Bluetooth.

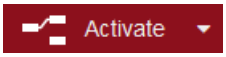
Enter the URL to broadcast

Double click the Eddystone node to enter the URL to broadcast:



Activate and broadcast your URL

Click the top-right "Activate" button.



A "Stopped" status appears on the eddystone node to indicate that it's actually not broadcasting.

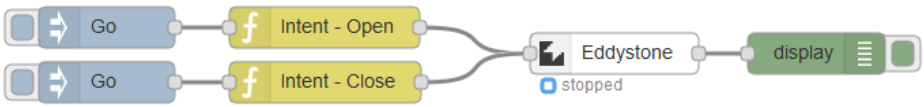
Then, click the "Go" button on the flow and the status on the Eddystone node becomes "broadcasting".

Access your hot deals thanks to the Physical Web

Use the Physical Web application (available on **Android** and **iOS**) to browse the broadcasted URL and look at the hot deals picture.

More

The second demo `PhysicalWeb demo 2` shows how to stop the broadcasting:



Credits

This instructable uses the (great) work

- of **Matthias Böhmer** (**https://github.com/matboehmer/node-red-contrib-eddystone**)
- and of **Don Coleman** (**https://github.com/don/node-eddystone-beacon**)

IFTTT Trigger

Use rules in the cloud with IFTTT ("if this then that").

IFTTT (**https://ifttt.com**) is a cloud service that allows users to create chains of simple conditional statements, called "recipes".

Discover recipes here: **https://ifttt.com/recipes**

All the recipes use channel that are created from the majority of social networks and inbox applications.

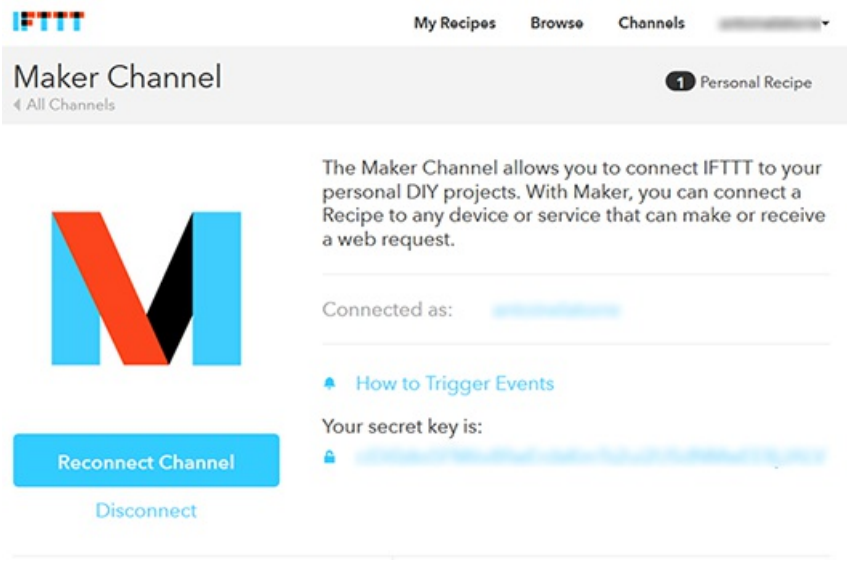
In order to generate IFTTT Trigger, we use the Maker channel that allows you to connect your app with the IFTTT service.

Before you start

An IFTTT account

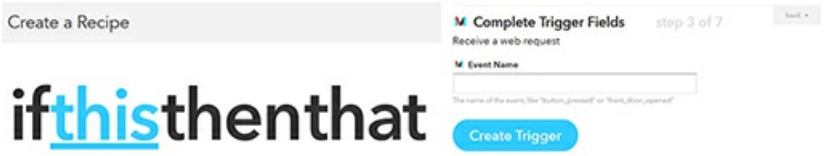
To use the IFTTT Trigger node you have to have an IFTTT account ! To create your account go on the IFTTT Web site **https://ifttt.com**.

After your inscription done, you have to connect your account with the **Maker channel**. Now you can create recipes with this channel ! A secret key have been created to link your account with the notifications that will be genarated, you have to remenber this key for later.



Create the recipe with Maker

Click on "My Recipes" link on the web site's top and "Create a Recipe". For "THIS" you choose the channel Maker and define an event name.



Finally for "THAT" you can choose only one channel to receive an email or a SMS on your phone (you can choose the channel you want).

Email configuration

Choose the "Email" channel to configure your notification. IFTTT use your account email by default.

SMS configuration

Choose the "SMS" channel to configure your notification. IFTTT demands your phone number and you have to type like that if you have, for example, a french number :

- 0036XXXXXXXXX (36 is the country calling code and the nine X

are your phone number without the first 0)

Configuration

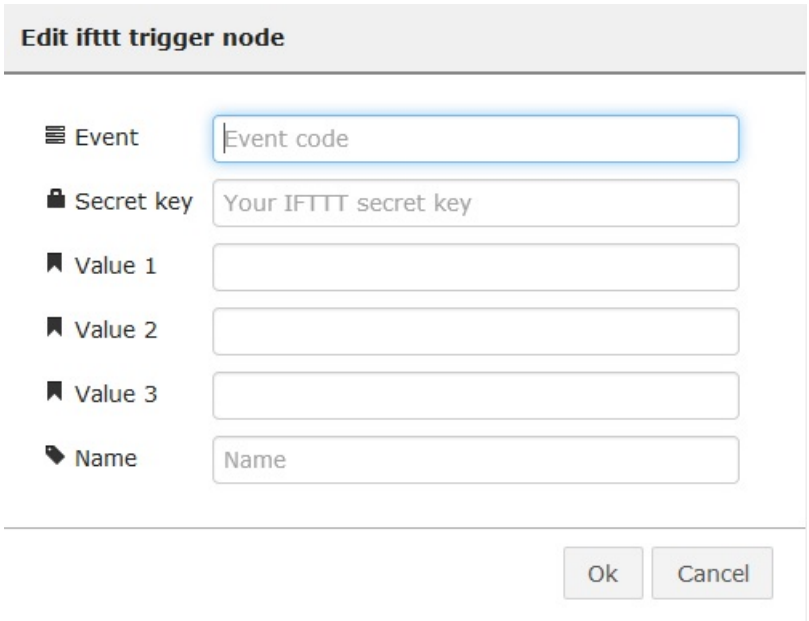
We will create a flow that will trigger some informations from the event you have created.

Follow these instructions :

- Drag and drop 1 go node from the nodes palette to your workspace,
- Drag and drop 1 IFTTT Trigger node from the palette to your workspace, and link it to the go node,

Once done, you should have this flow :

By double clicking on the IFTTT Trigger Node, you will see the configuration panel.



Then, you have to set some values :

- `Event` : the event name of your previous recipe,
- `Secretkey` : your secret key,
- `Value1`, `Value2` & `Value3` : three optional values

Click Deploy button.

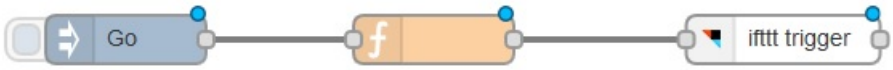
Usage

By clicking on the go node, you will send the data to the event and IFTTT will redirect the notification in terms of your recipe.

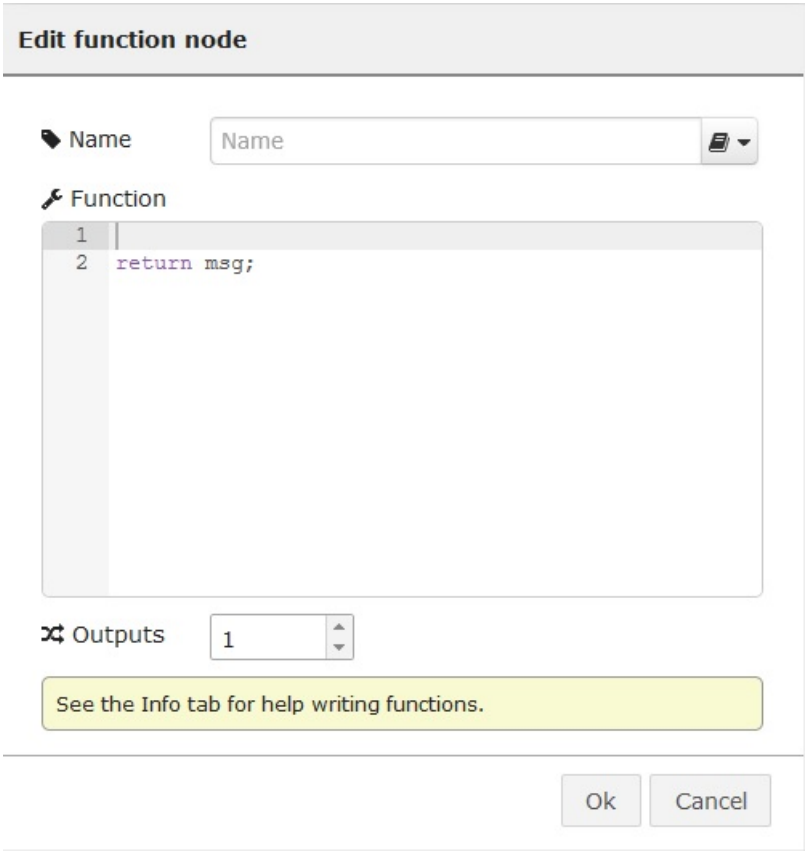
Going further

We will create a flow that will trigger some informations to the event you have created.

Drag and drop and wire one Go node, one Function node and one IFTTT Trigger node to build this flow :



By double clicking on the Function Node, you will see the configuration panel.



Then, you have to set some values :

- `msg.event` : the event name of your previous recipe,
- `msg.secretkey` : your secret key,
- `msg.value1`, `msg.value2` & `msg.value3` : three optional values

The function might be this :

```
msg.event = "front_door";
msg.secretkey = "your secret key";
msg.value1 = "Open";
```

```
return msg;
```

Click activate button.

Control Hue bulbs

Introducing Philips Hue bulbs

The hue bulbs can be turned on and off, and change their color dynamically using the Hue API. The ThingBox provides a simple access to every control you can have on your hue bulbs.

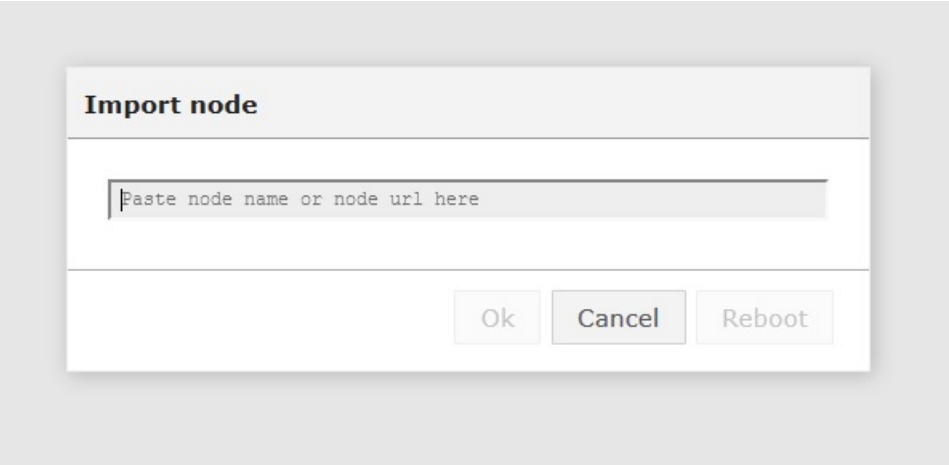
Before you start

The Hue node we will use in this tutorial will only work if you have already configured your Hue Bridge and added some Hue bulbs to it.

For more informations about how to do it, go **on the Phillips Web site**.

Configuration

First off all, you have to install the `ttb-node-hue` via the Import->Node menu. You will see this window :

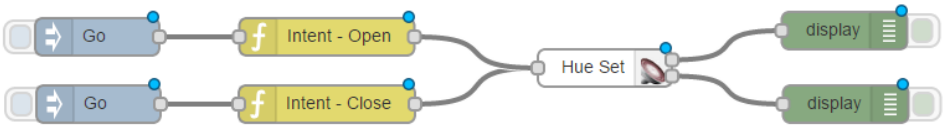


To install it, you have to write `ttb-node-hue` in the form of this window. Click `OK` button, wait a moment for the installation and when the `Reboot` button can be clicked, click on it !

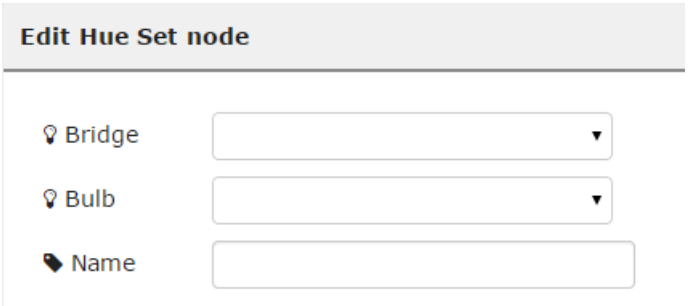
Build a flow

Now, we will create a flow that will allow us to switch on and off a hue bulb.

Drag and drop nodes to your workspace and wire them to build this flow :

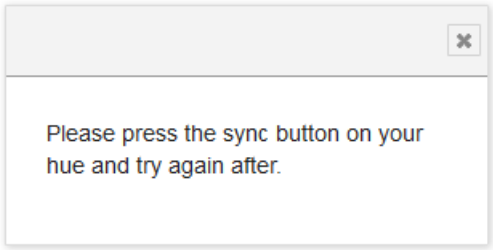


By double clicking on the Hue set Node, you will see the configuration panel.



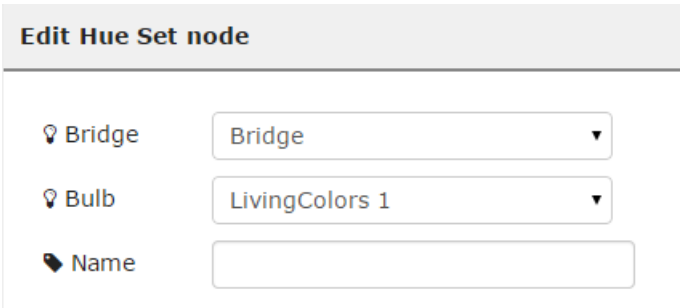
Then, you have to select a bridge in the bridge list. Hue bridges are automatically detected by the node.

You will see a popup that will explain you that you have to push your Hue bridge Sync button (the one in the middle) in order to continue.



Close this popup and go find your Hue bridge, and press the round shaped button (the one at the middle of the device)

Once it is done, you have to open quickly the Hue node on your browser and there will be the list of the Hue bulbs that are already linked to your Hue bridge.



Select the one you want to take a try with. Click the OK button, and you are now done with the Hue node configuration. Click `Activate` button.

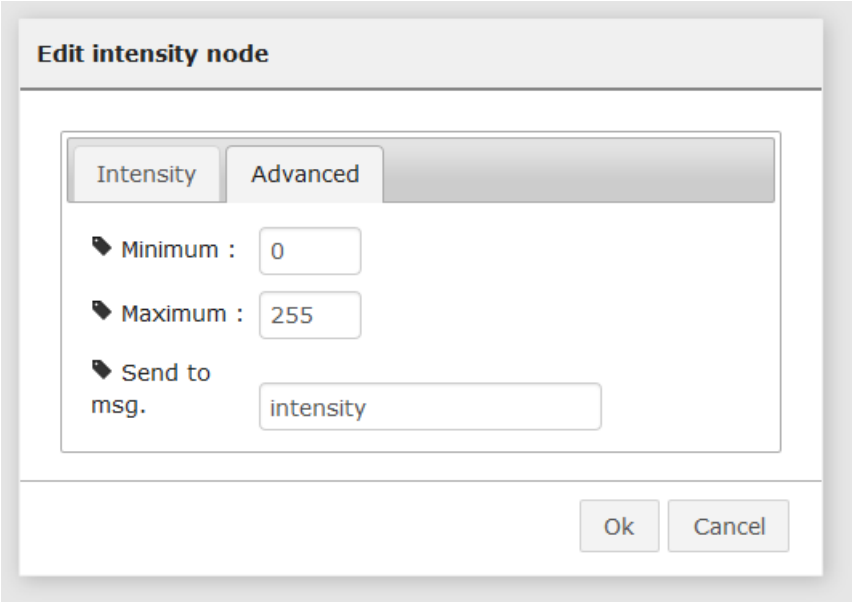
Usage

By clicking on the first go node, you will switch your light on, and you can switch it off by clicking the second go node.

Change the bulb intensity

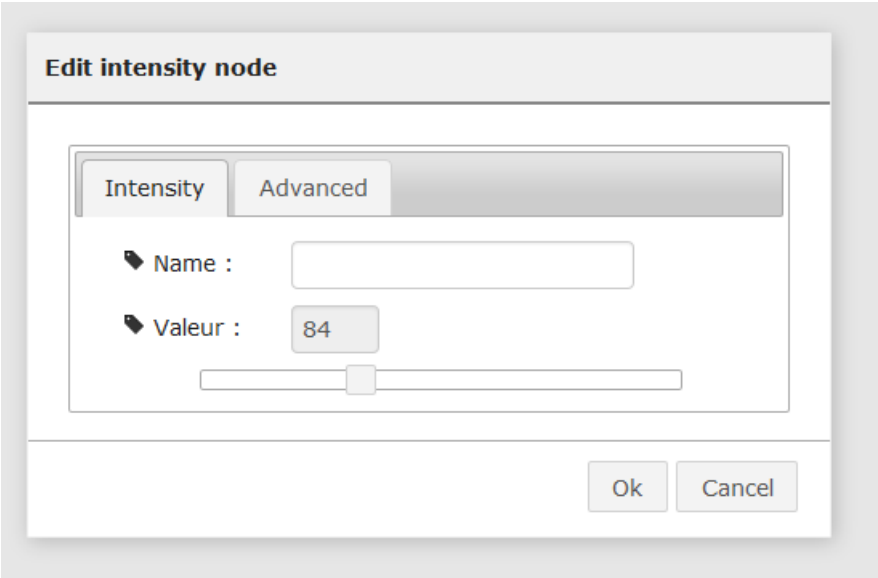
In this flow, we add an intensity node:

This node sets the intensity of the flow. The default allows a value between 0 and 100, but the Hue can go til 255. This can be set in the node parameters:

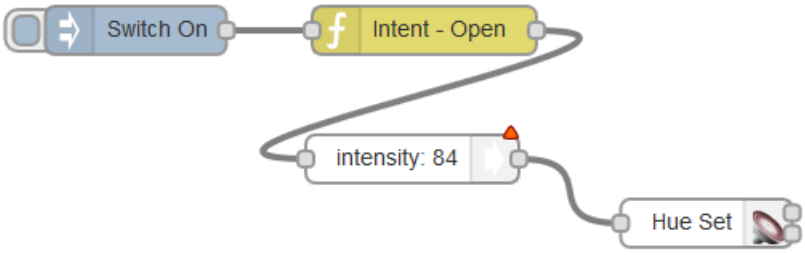


In this dialog, just change the Maximum to 255.

Then, you can edit the value in the first tab:



You get the following flow:

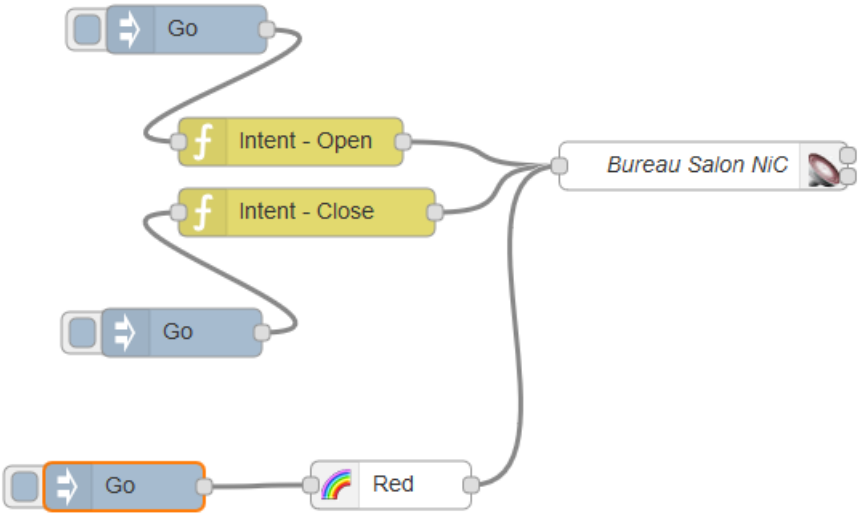


Change the bulb color

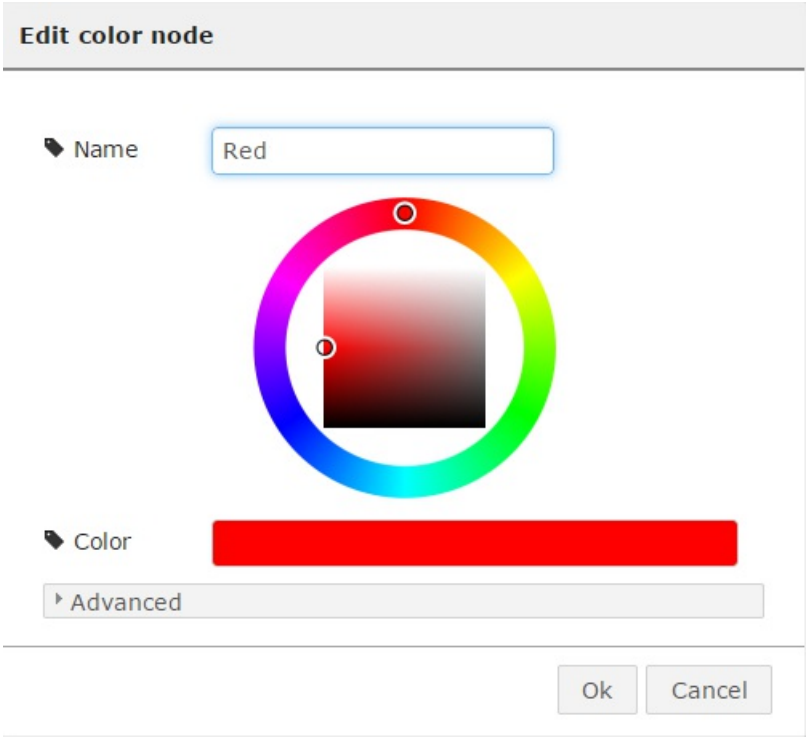
Drag'n drop a color node from the palette:

To configure to color node, double click on the node and pick with

your mouse the color you like in the colored circle.



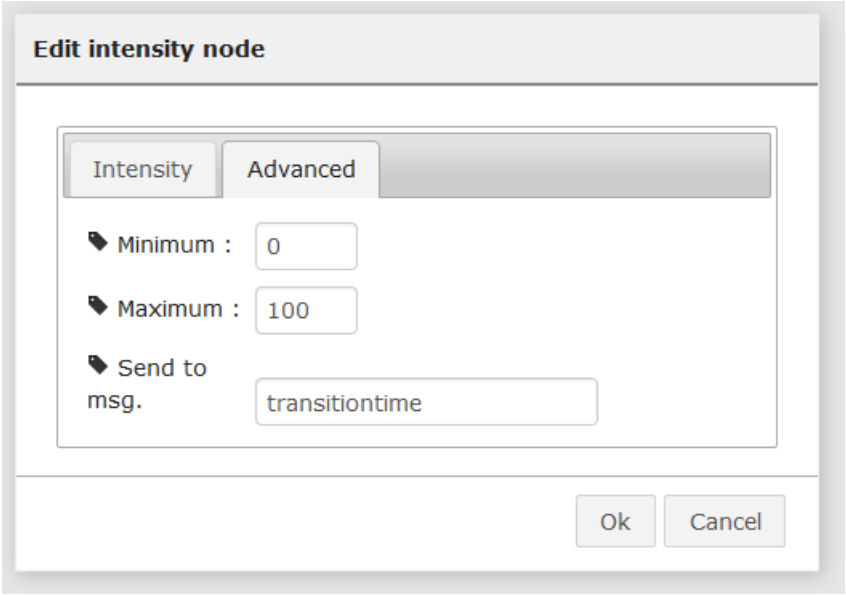
You can rename your node with the color you choosed :



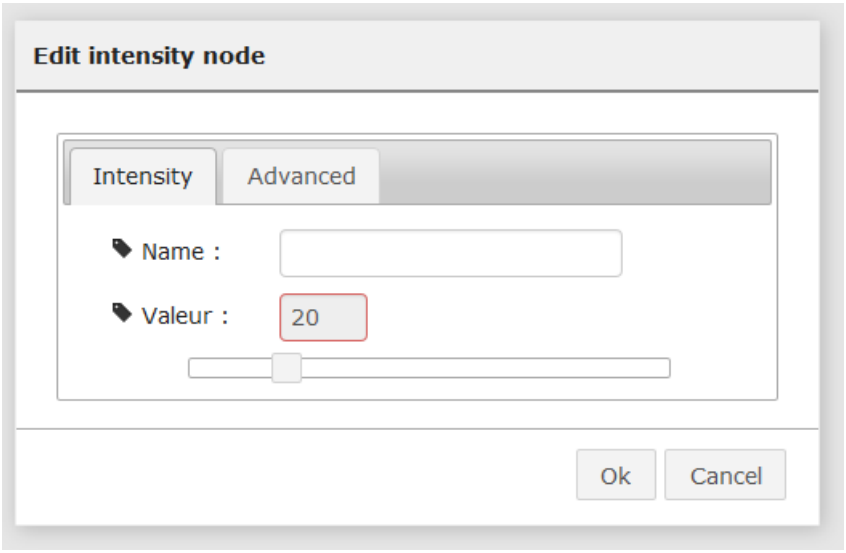
When all the configuration is done, click on the `Activate` button again.

Add a transition time

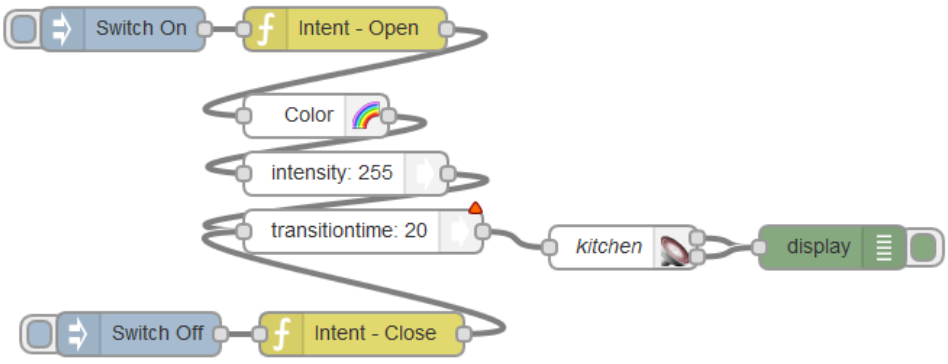
To change the transition time, we use the intensity node again, but we need to change the parameter of the flow from "intensity" to "transition time":



you can now change the value in the first tab:



Here is a typical flow:



Usage

Now, by clicking on the go node, you can dynamically change your Hue bulb color.

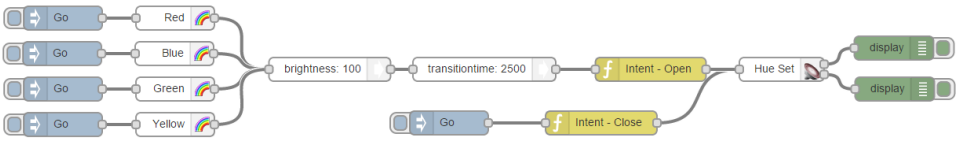
You can also adjust the brightness and the transition time by

sliding the values in the intensity nodes configuration bars.

As in the first example, the last go node turns the light off.

You are now done with Hue flows on theThingBox !

You can even make a more complex flow, that makes the color change by clicking on the different go nodes :



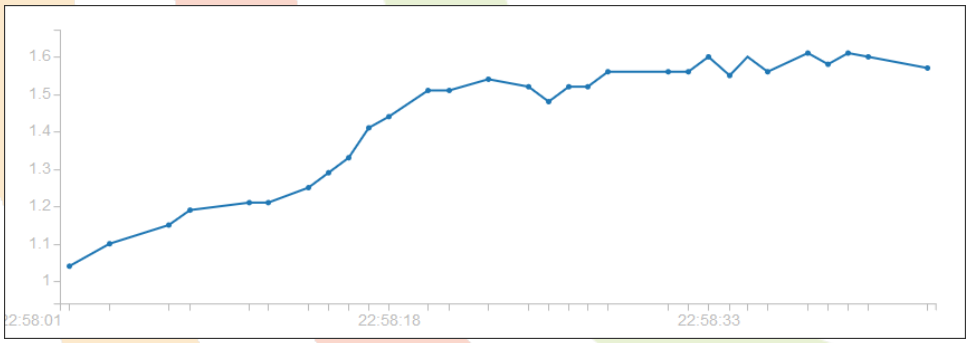
Credits

Adapted work of **node-red-contrib-hue** from **urbiworx.de** based on **Peter Murray hue lib**.

Charts in cloud with IBM's IoT

This tutorial shows how to build a simple chart on the cloud with a simple node that sends values to the cloud.

Here how the result looks like:

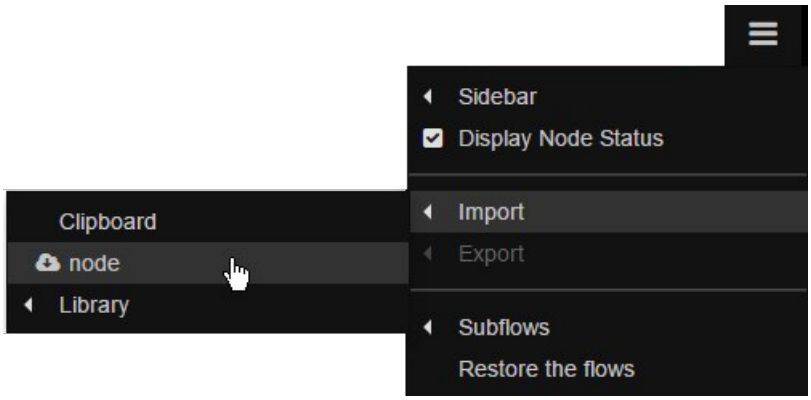


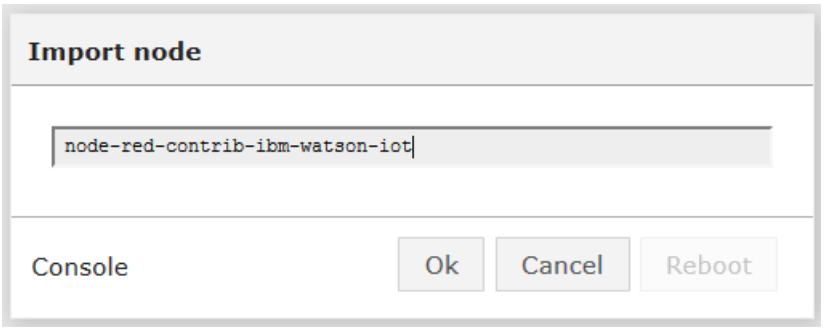
For this tutorial, we use the IoT Bluemix cloud from IBM and a node also provided by IBM to send datas.

IBM offers a GREAT idea named "QuickStart" where you can get a result without all the pain of registering the service. First give a seamless result to the customer, then if he wants more you can ask him to invest some time (or money). Great lesson a lot of cloud service providers should ear.

Install the nodes

Click the right menu in the Node-RED web page. Choose `Import` then `node`.





Type

`node-red-contrib-ibm-watson-iot`

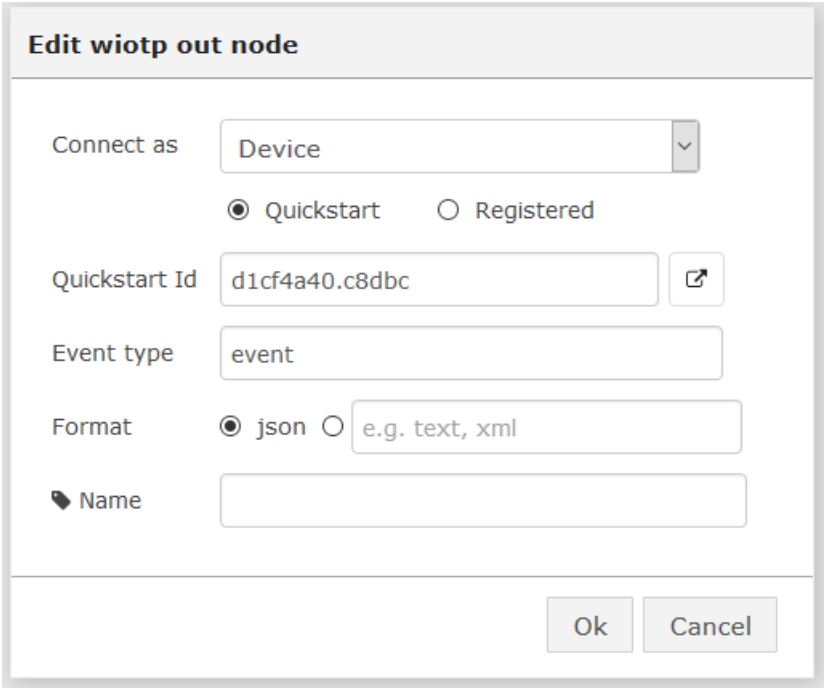
in the edit box and click the OK button.

This imports a third party software. The installation may take minutes (a progress bar shows the progress), depending on what is already installed on your device.

At the end, click the `reboot` button to restart the device.

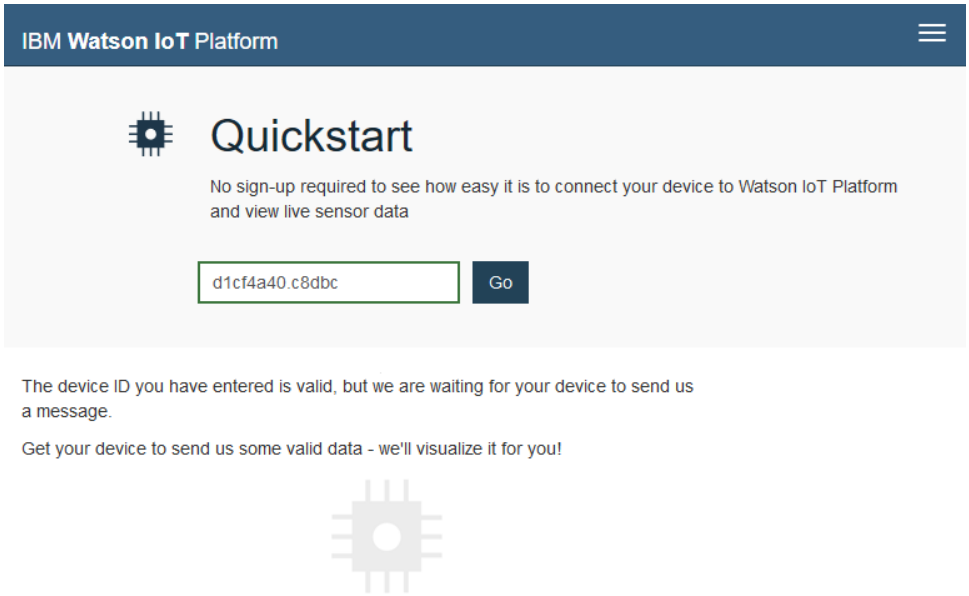
Now you can find this node in the palette (use the filter to find it):

Drag'n drop it on the workspace, and double click it to set parameters:



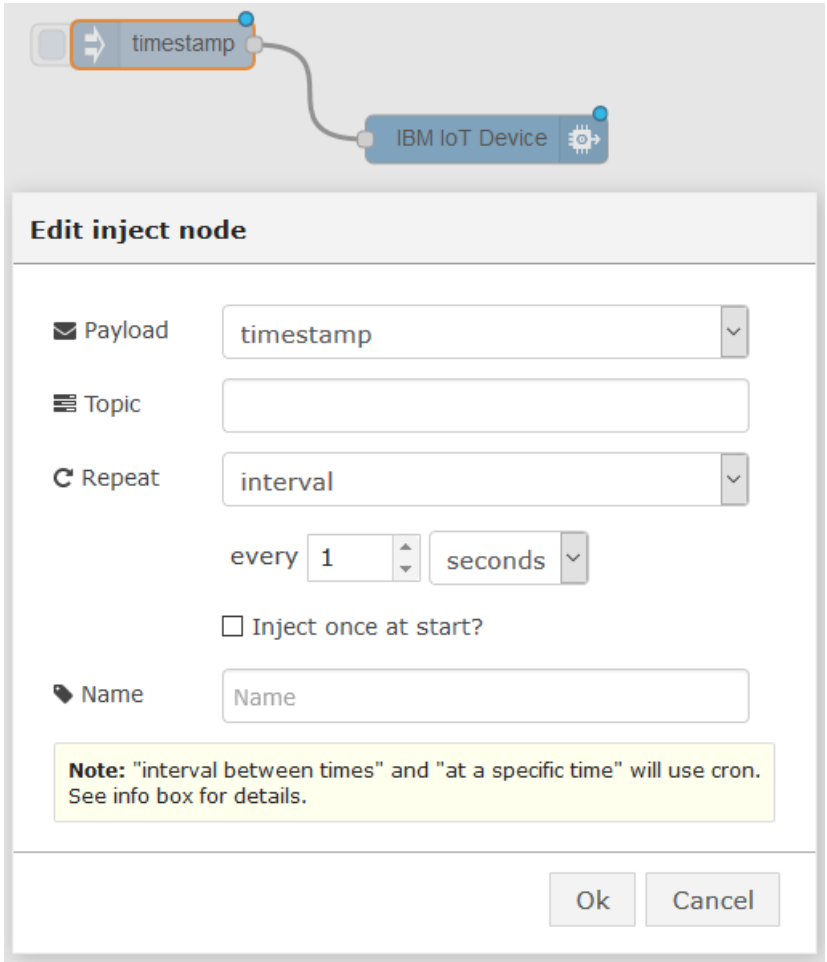
You can click the button below to get the cloud bluemix chart:

all these technical datas should normally not appear in a quickstart node that provide a seamless user experience.

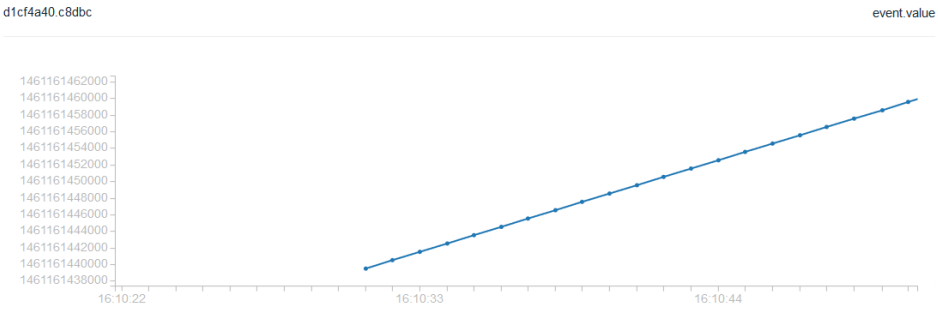


Send data and build a chart

Now you can send any numeric value. Let's build the simpliest flow: add an "inject" node and set it to send the current date as a number every one second:



And look the beautiful result:



Beyond this Quickstart mode, IBM offers a full featured cloud for IoT.

Credits

This tutorial uses IBM IoT Bluemix: **http://discover-iot.eu-gb.mybluemix.net**

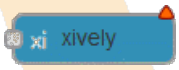
Where to go now

- Learn more about IBM IoT Bluemix: **http://discover-iot.eu-gb.mybluemix.net**
- Get a full documentation on the Node: **https://www.npmjs.com/package/node-red-contrib-ibm-watson-iot**
- Discover other IoT cloud offers with the "**Use IoT platforms**" tutorial.

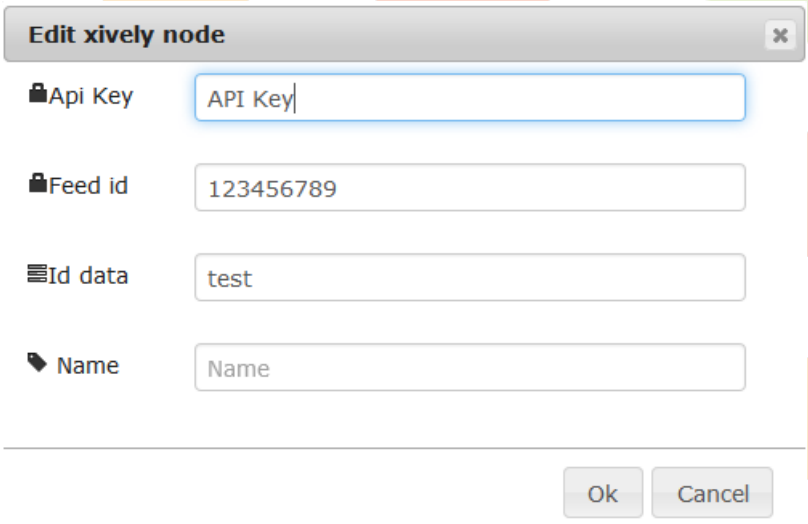
Use IoT platforms

For all the following node, create an account on the matching website. Datas created are only viewable on website. the vocabulary used is the same on their site. Some informations are available in the section information on Node-RED for each node.

Node Xively (https://xively.com)

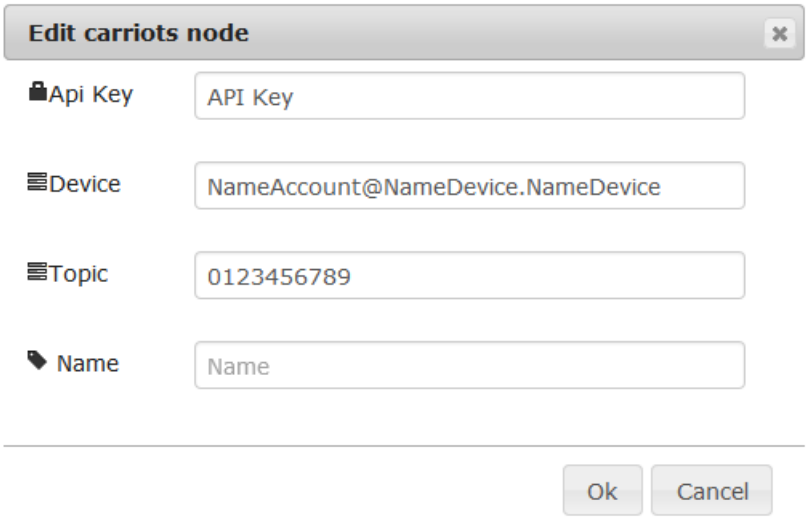


This node allow the user to create some graphs that display data with several display options. It requires the API Key of the account, the feed ID and the name of the data.



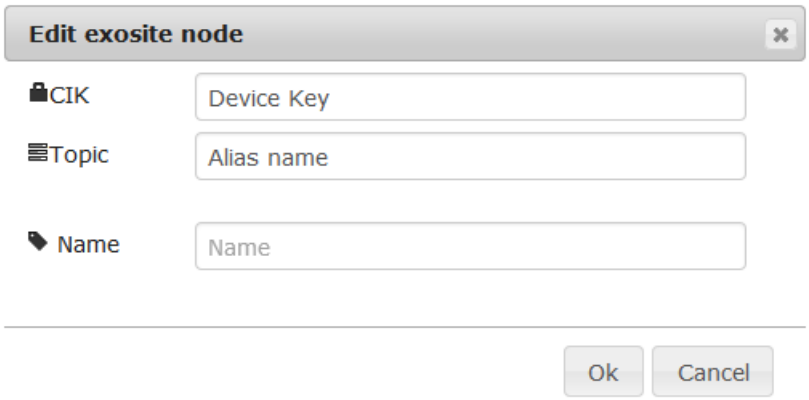
Node Carriots (https://www.carriots.com/)

This node allows the user to have a mailbox in which he can store all types of data. It requires the full privileges API Key of the account, the device description of your device and an optional topic.



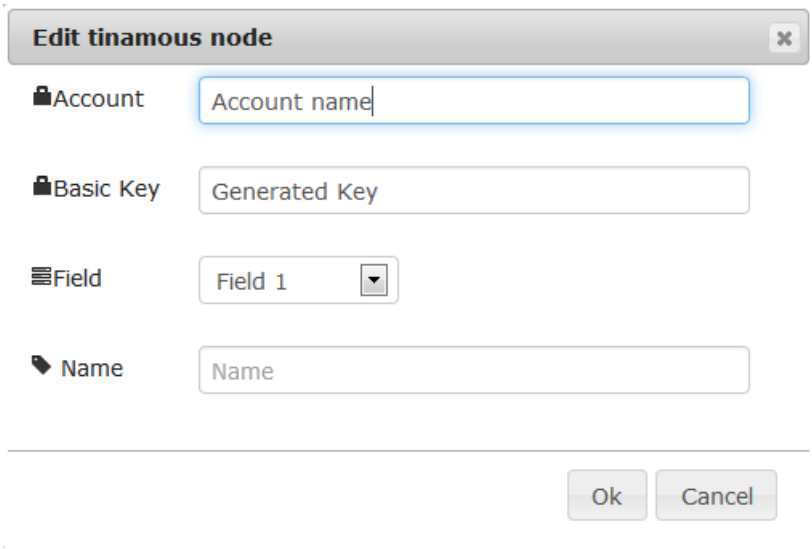
Node Exosite (http://exosite.com/)

This node allows the user to send and store data sent about a created device. It requires the CIK of the created device and the alias name of the created data.

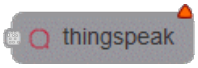


Node Tinamous (https://tinamous.com/)

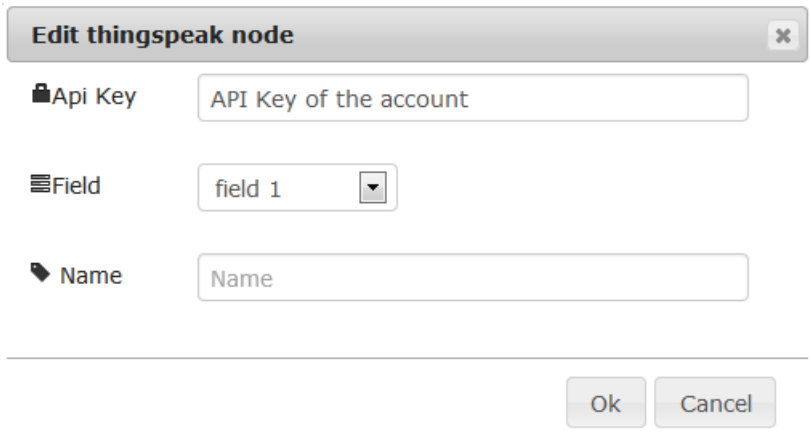
This node allow the user to have 12 graph for each device. Each graph can be download in jpeg, png, pdf or svg format. It requires the name of the account, the header generated in the settings of the created device.



Node Thingspeak (https://thingspeak.com/)

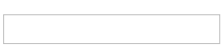


This node allows the user to have 8 graphs. Data of graphs can be import/export in csv files. It requires the API Key Write of the account and the number of the graph.



Node Ubidots (http://ubidots.com/)

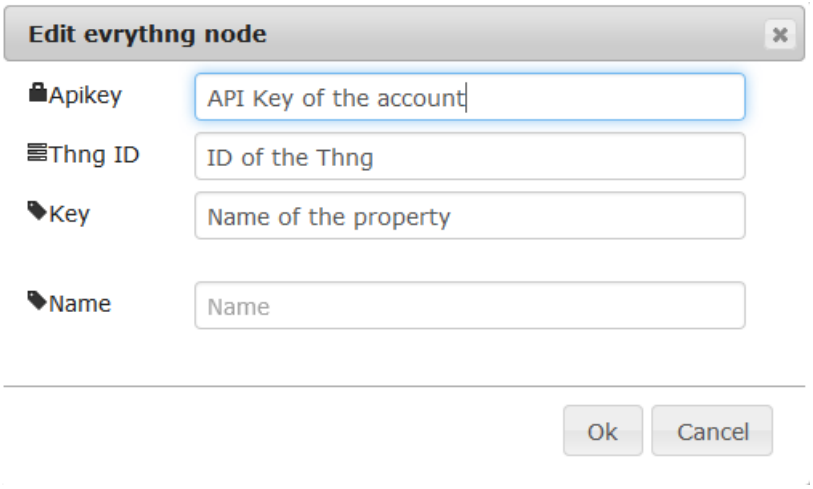
This node allows the user to have graphs. They can be export to csv format. It requires the token authentication of the account, the ID of the created variable on the site.



Node Evrythng (http://evrythng.com/)

This node allows the user to create properties for a Thng and give

them a value. These properties can be seen thanks to graphs. In the site you can create products and Thng that offer the possibility to organize data. It requires the API Key of the account, the ID of the created Thng and the name of the property of the Thng.



Read the cloud !

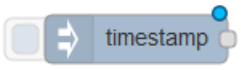
Receive an email when the Bitcoin value changes

This tutorial is a complete example showing how to get a value from the cloud and email it.

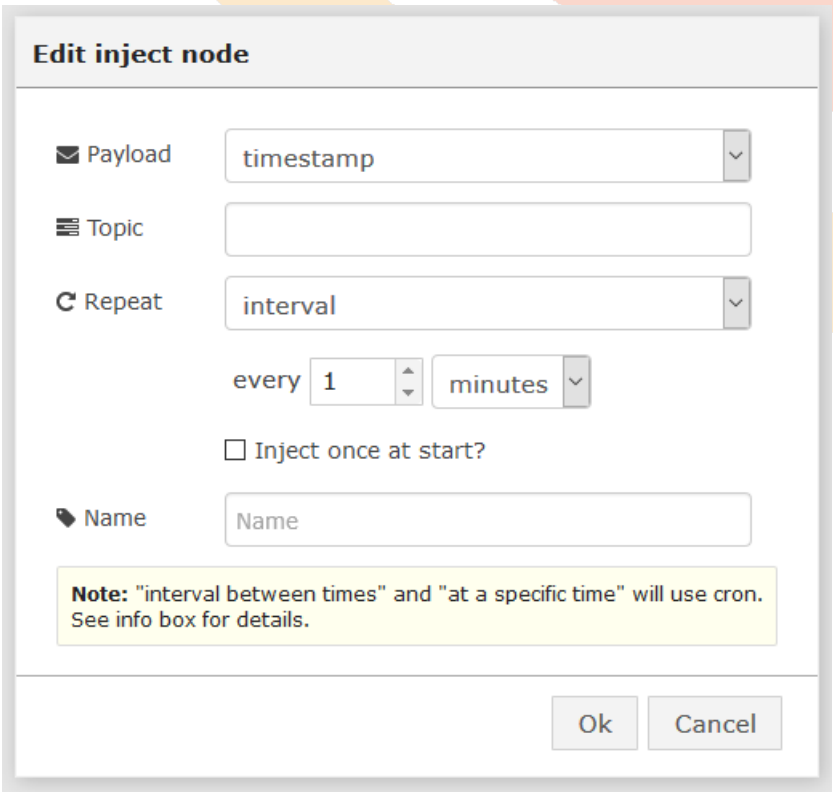
This example gets the bitcoin value every minutes and sends an email when the price of bitcoin has a significant change.

Inject

First, add an inject node and edit it to repeat the injection every minute.

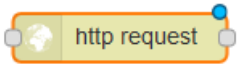


Double click it to edit its values.



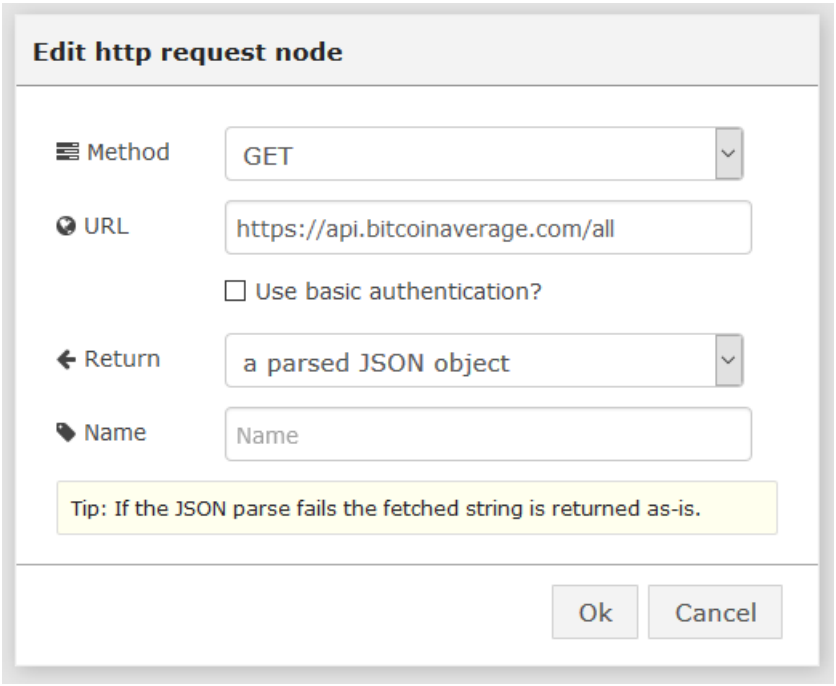
Request bitcoin value

Next, add httpRequest node in the section `functions` :



To request the following address:

- https://api.bitcoinaverage.com/all



Compare with the last value

Now, add a function node which takes only the price of bitcoin and compare with the last value.

To do this, the value is saved in a context variable at the beginning and the flow sends an email if there is a big difference (100 $).

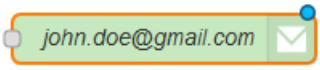
```
// Create the bit coin context
context.bid = context.bid || 0;

var obj   = msg.payload;
var rates = obj.USD.global_averages;
var bid   = rates.bid;
//var ask = rates.ask;

if ((bid >= (context.bid + 100)) || (bid <= (context.bid - 100))){
    var Newmsg = { payload: "ALERT new bid : "+bid };
    context.bid = bid;
    return Newmsg;
}
```

Send an email

Add an email node:



Add to whom send the email, the user email address with his password and his service.

The final flow

This is the final flow :

Flash a SD Card

TimeSquAir can be updated by flashing a new SDCard.

Two steps:

** Get the binay file of the SDCard from the **http://TimeSquAir.io** web site ** Unzip and write this file to a SDCard

Some tools are needed:

- A SD card. On the Raspberry Pi computer, all the software is on a SD Card. Thus the first step is to write the firmware (the software of the device) to the SD card.

Warning : the SD card will be totally over-written, so this SD card will be used only for the Raspberry Pi. Don't forget to backup your Node-RED files if you update

- A Windows or Macintosh computer (with a SD port to access the SD card)
- The firmware,
- A particular application for the computer to write the SDCard.

Download the firmware

The firmware is a very big file (many Gb) that is downloaded from the cloud and then written on the SDCard that will then be inserted in the Raspberry Pi. As this firmware runs on the Pi, it is the same if you use Windows or Macintosh.

Download the image from the TimeSquAir when site, using the Download menu.

Unzip it

This part depends on your computer.

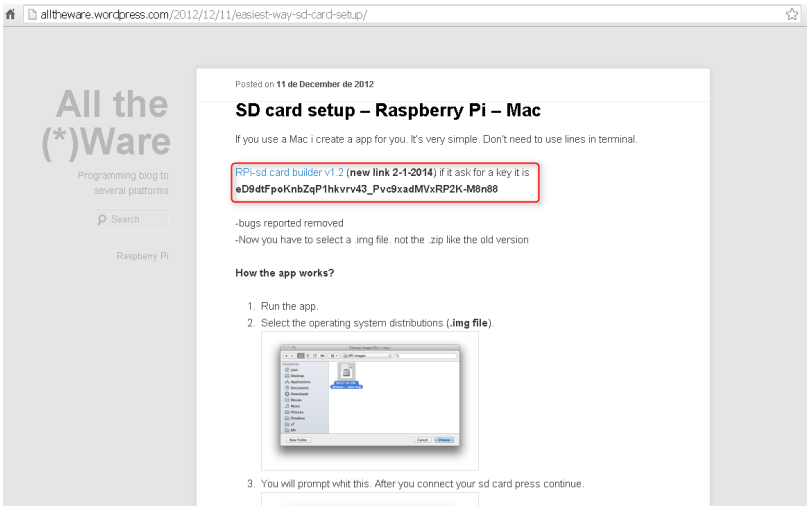
Download the application to write the firmware on the SD card

On **Windows**, download the application **Disk Imager**:

- **http://sourceforge.net/projects/win32diskimager/files/Archive/Win32DiskImager-0.9.5-binary.zip/download**

On **Macintosh**:

- **http://alltheware.wordpress.com/2012/12/11/easiest-way-sd-card-setup/**

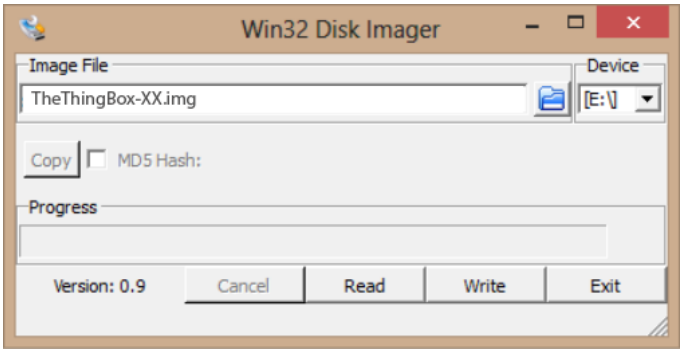


Write the firmware to the Pi SD card

A common mistake is to flash the zip file to the SDCard: it should be unzipped first !!!

On Windows

- 1 : Extract the content of the firmware from the zip file (It works well with **Bitser** or **WinRAR** on Windows and Stuffit on Mac but other zip software should work **but be aware that the generic unzipper provided with Windows/Mac does not extract the file for an unknown reason**!!). We try to zip another way for next release.
- 2 : Extract the content of the SD Disk Imager from the zip file
- 3 : Put the SD card into the SD port of the computer
- 4 : Launch the Disk Imager application
- 5 : Choose the unzipped firmware
- 6 : Choose the destination Device to be the SD Card and **NOT** any other hard drive



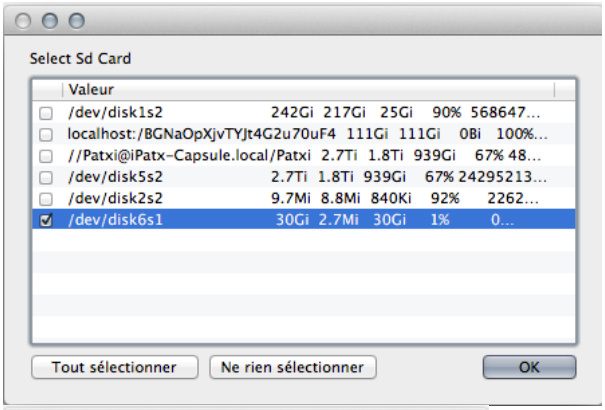
- 7 : Choose Write and be patient

Finally, the SD card is ready to use.

On Macintosh

- 1 : Extract the firmware and the rpi-sd card builder.zip

- 2 : Launch SD card builder
- 3 : Select the firmware to write
- 4 : Choose the SD card on which to flash the Pi image, confirm and enter the password if requested

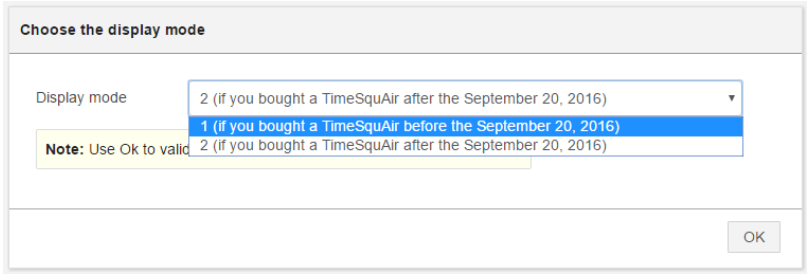


- 5 : SD card setup is flashing the Pi image on the SD card. Be patient

Finally, the SD card is ready to use.

Settings

If you bought your TimeSquAir before 20 september 2016, you have to go to the settings page (using the right menu, then settings), and choose the display mode 1:



TimeSquAir

www.timesquair.io

By DIGITALAIRWAYS™