

Introduction to Node-RED

Hackerspace Charlotte

Lisa Waugh

Install Firefox

- *Note:* the default browser included in Raspbian, Epiphany, has some quirks that mean certain keyboard short-cuts do not work within the Node-RED editor. We **strongly** recommend installing the Firefox-ESR browser instead:
- `sudo apt-get install firefox-esr`

What is Node-RED?

- Node-RED is a tool for wiring together hardware devices, APIs and online services in new and interesting ways
- Node-RED provides a browser-based flow editor that makes it easy to wire together flows using the wide range nodes in the palette. Flows can be then deployed to the runtime in a single-click.
- JavaScript functions can be created within the editor using a rich text editor.
- A built-in library allows you to save useful functions, templates or flows for re-use.

- The light-weight runtime is built on Node.js, taking full advantage of its event-driven, non-blocking model. This makes it ideal to run at the edge of the network on low-cost hardware such as the Raspberry Pi as well as in the cloud.
- With over 225,000 modules in Node's package repository, it is easy to extend the range of palette nodes to add new capabilities.
- The flows created in Node-RED are stored using JSON which can be easily imported and exported for sharing with others.
- An online flow library allows you to share your best flows with the world.

Starting Node-RED on Raspberry Pi

- Click on the Raspberry in top left corner
- Select Programming
- Select Node-RED
- A command window will open and Node-RED console will appear
- Open Firefox
- localhost:1880

Node-RED Canvas

The screenshot displays the Node-RED web interface. At the top, there's a black header with the Node-RED logo and a 'Deploy' button. Below the header, a tab bar shows 'Flow 1', 'Flow 2', 'Flow 3', and 'Flow 4', with 'Flow 4' being the active tab. On the left, a sidebar titled 'input' lists various nodes: inject, catch, status, link, mqtt, http, websocket, serial, tcp, mqlight, and ibmiot. The 'inject' node is highlighted, and a tooltip is visible over it. The tooltip contains the following text:

inject

Pressing the button on the left side of the node allows a message on a topic to be injected into the flow.

On the right side of the interface, there's a panel with 'info' and 'debug' tabs. The 'info' tab is active, displaying detailed information about the selected 'inject' node. The information includes:

- A description: "Pressing the button on the left side of the node allows a message on a topic to be injected into the flow."
- Default payload: "The payload defaults to the current time in millisecs since 1970, but can also be set to various other javascript types."
- Repeat function: "The repeat function allows the payload to be sent on the required schedule."
- Inject options: "The *Inject once at start* option actually waits a short interval before firing to give other nodes a chance to instantiate properly."
- Flow and Global options: "The *Flow* and *Global* options allow one to inject a flow or global context value."
- Note: "Note: 'Interval between times' and 'at a specific time' uses cron. This means that 20 minutes will be at the next hour, 20 minutes past and 40 minutes past - not in 20 minutes time. If you want every 20 minutes from now - use the 'interval' option."

First Flow

The screenshot displays the Node-RED web interface. At the top, the 'Node-RED' logo is on the left, and a 'Deploy' button is on the right. Below the logo is a search bar labeled 'filter nodes'. The main workspace shows 'Flow 4' selected, containing a 'timestamp' node connected to a 'msg.payload' node. On the left sidebar, under the 'output' category, the 'debug' node is highlighted. On the right sidebar, the 'debug' tab is active, showing a log entry: '1/11/2017, 5:09:02 PM 4aa32248.d272fc' followed by 'msg.payload : number' and the value '1484172538874'. The 'debug' tab is highlighted with a red box.

Alter Flow 1

The screenshot shows the Node-RED web interface. On the left, the 'filter nodes' search bar is at the top, followed by a list of nodes including 'mqlight' and 'ibmiot' under the 'output' category. The main workspace is titled 'Edit inject node' and contains fields for 'Payload', 'Topic', 'Repeat', and 'Name'. A dropdown menu is open for the 'Payload' field, showing options: 'timestamp', 'flow.', 'global.', 'string', 'number', 'boolean', and 'JSON'. A yellow note box at the bottom of the dialog states: 'Note: "interval" and "at a specific time" will use cron. See info box for details.' The right sidebar has tabs for 'info' and 'debug'. The 'debug' tab is active, showing a log entry: '1/11/2017, 5:09:02 PM 4aa32248.d272fc msg.payload : number 1484172538874'. The top right of the interface has a 'Deploy' button and a menu icon. The bottom status bar shows the URL: 'https://lrmwiotstarter.mybluemix.net/red/#'.

Node-RED

filter nodes

Flow

Edit inject node

Cancel Done

Payload

Topic

Repeat

Name

timestamp

flow.

global.

string

number

boolean

JSON

Note: "interval" and "at a specific time" will use cron. See info box for details.

info debug

all flows current flow

1/11/2017, 5:09:02 PM 4aa32248.d272fc
msg.payload : number
1484172538874

https://lrmwiotstarter.mybluemix.net/red/#

Alter Flow 1

The screenshot shows the Node-RED web interface. On the left, the 'output' category is selected in the node palette, showing various output nodes like 'debug', 'link', 'mqtt', 'http response', 'websocket', 'serial', 'tcp', 'udp', and 'mqlight'. The main workspace displays the 'Edit inject node' dialog. The 'Payload' field, which includes a character selection dropdown (showing 'a' and 'z') and a text input containing 'Hello Hackerspace!!', is highlighted with a red rectangular box. Below the payload field, the 'Topic' field is empty. The 'Repeat' dropdown is set to 'none', and the 'Inject once at start?' checkbox is unchecked. The 'Name' field contains the text 'Name'. At the bottom of the dialog, a yellow note states: 'Note: "interval between times" and "at a specific time" will use cron. See info box for details.' To the right of the dialog, the 'debug' console is open, showing a message from '1/11/2017, 5:09:02 PM' with a unique ID '4aa32248.d272fc'. The message content is 'msg.payload : number' followed by the value '1484172538874'.