Programming a Shakemeter in Node-Red

Shake into the world of programming and computer science with Node-Red!

Introduction to Programming with Node Red

Raspberry Shake processes its own data, and publishes it to its own public databases. However, it is also possible to configure the Shake to send a raw data-stream from itself to any device your network can reach. Raspberry shake uses UDP, a type of data transfer protocol, to send data from itself to any list of IP addresses and ports that happen to be listening. Using this raw data stream you can create your own graphs, or set up alerts and triggers in Node-Red.

This means that your Shake could be physically located anywhere in a building (or—using the proper networking setup—anywhere in the world) and you can still comfortably receive the data in Node-Red on your own computer. Imagine the possibilities! An email alert being sent-out by your Raspberry Pi at home, because your Raspberry Shake at school detected some unusual vibrations… Or you could trigger a camera to snap and email you a picture whenever footsteps are detected in your room. The possibilities are endless!

What is Node-Red?

Node-Red is a visual, flow based programming language powered behind the scenes by Javascript. It is an excellent way to introduce students to programming concepts and basic programs without scaring them with text based languages like Python and Javascript. It utilizes basic input, output, and function nodes to create “flows”, which can then be executed as programs. It is similar to the popular educational programming language Scratch, but much more applicable in real-world situations that incorporate hardware and wiring.

The most simple way to run a Node-Red program is using a Raspberry Pi. It is already installed natively on the Pi, so there are no extra downloads or terminal commands necessary to run the server. If a Raspberry Pi is not available, you can still run and use Node-Red on any laptop or desktop, it is just slightly more complicated to set up. To set up Node-Red on a device other than a Raspberry Pi, click here.

Once you start running the local server, you can access the browser-based interface by entering “localhost:1880” (you can also enter 127.0.0.1:1880, they are synonymous) into the browser of the same device running Node-Red. You should arrive at the Node Red workspace (Figure 1).

Figure 1: Node-Red flow

Node-Red has exceptional functionality pertaining to Raspberry Shake and GPIO pins, as well as website and networking capabilities. It also has the ability to send and receive emails and tweets. Another powerful feature we will be using often in our projects is the Dashboard function, which is a locally hosted graphical user interface (GUI) that can contain everything from graphs and text to interactive buttons. Our ShakeMeter (that you will soon learn how to make) is powered by the Dashboard GUI.

What is UDP, and how it it used with RS?

When data is exchanged over the internet, it is sent using a protocol, which is a way of organizing data so that both parties know what’s going on. If you made an agreement with a friend to always use red envelopes when a message was urgent, that would be an example of a protocol. Most of the time when you download a file or send an email it is sent using TCP protocol; which is rather strict. For example, if something messes up and the TCP packets (analogous to envelopes with letters inside) arrive out of order, they will be reorganized and only sent to the destination when everything is sorted out. TCP also requires acknowledgements for every piece of communication sent, by both the sender and the receiver. TCP acknowledgements are kind of like asking sending a letter to your friend and asking "hey did you get my letter?" and when they say "yes," you say "OK."

UDP is much less strict, and is more like a one-way firehose of data rather than an orderly sequence of letters with acknowledgements and replies. If UDP packets arrive out of order, or have some missing, or even if the data inside is unreadable or even lost along the way, there is no acknowledgement, and no re-sending of lost or bad data. This might sound like a bad thing, but it can make communication very fast inside a local network, where data very rarely gets lost or garbled. Because it is very fast and requires no acknowledgments, UDP also has a feature called multicast, which allows a data stream to be sent to multiple destinations at the same time. Raspberry Shake makes this powerful feature easily accessible for users to configure and use.

Connecting your Raspberry Shake to Node-Red

In order to start the process of connecting your Raspberry Shake to Node-Red, you must make sure your Raspberry Shake and the device running Node-Red are on the same network. Then, find the local IP address of the the Node-Red device. You can do this by typing the command "ifconfig" (linux) or "ipconfig" (Windows) into the Node-Red device's Terminal to find your local IP. It may be difficult to find, make sure you get the right one.

Once you have the local IP address of the device running Node-Red, you can set up the UDP stream from the Raspberry Shake. Watch the video above or follow the step-by-step instructions below to get started!

Step 1: Connect to your Raspberry Shake's web front end (rs.local) and go to settings

Step 2: From the Settings page, click on the UDP STREAMS tab

Step 3: Enter the IP address of the "listening" computer that is running Node-Red, and the port that that the data is sent to. We recommend using default port 8888. Remember, your IP address will be different than the one in the photo

> *Important:*
>
> - *Both devices, the Shake and the "listening device", must be on the same local network*
> - *Remember to click both the "plus icon" button and the "Save" button when adding the UDP stream*

Reading the stream on Node-Red:

Back in Node-Red: starting with a fresh flow, we'll drag the UDP IN node in.

For the configuration chose the same port as we used in the rs.local config, in my case that’s 8888, and make sure to set output to ‘a String’ instead of ‘Buffer’.

At this point, if you hook your UDP RECEIVER node up to a debug node you should already be able to see a stream of data coming in. If not, something is wrong with your connection (probably, the Raspberry Pi isn’t on the same network subnet as the Raspberry Shake).

Next, we need the Raspberry Shake Parser node, made by the Raspberry Shake team specifically to process it’s UDP data.

This can be installed by going into the palette manager. As of August 2019, it’s the only result when you search ‘shake’.

The installation may take a minute or so, and then you can chain it with the UDP node like so.

The Rshake Parser node takes the raw output by the UDP node, and turn it into an object with various data all sorted into properties. Using a debug node, we can examine the output. For example, the ‘channel’ property allows you to discriminate between different sensors on the shake. We are using a Raspberry Shake 1D, so we only get one channel ‘EHZ’.

Now congratulations! You have completed the first step of connecting. Continue below to learn how to process the data and make a Shakemeter!

Instructions

(PDF download)

Making a Shakemeter in Node-Red

After following the first tutorial on connecting the Shake to Node-Red, now we can start to use Node-Red to process and visualize the seismic data that is being streamed through the UDP port. The first thing we will do with the data is create a “Shakemeter”, a program that takes the real-time seismic data and converts it into a live visual measurement of the absolute amplitude and velocity at which the ground is shaking!

Watch and follow along with the video tutorial below or follow along with the text instructions to program yours!

Make sure that there is UDP data being received by checking the debug tab. As you can see, the data coming through the Rshake Parser is still in object form, consisting of multiple types of information. In order to separate the numerical data and to quantify the shaking, another function node is needed to process the data. Below is an example of a node we made to address this.

Using the data:

[{"id":"d5d2efbb.2600e","type":"function","z":"b308f371.c9fa","name":"Absolute Amplitude","func":"acc = 0;\nnum = 0;\nfor(const p of msg.payload.packets)\n{\n acc += Math.abs(p);\n num++;\n}\n\nacc /= num;\n\nmsg.payload = acc;\nreturn msg;","outputs":1,"noerr":0,"x":570,"y":360,"wires":[["c8911fad.47f72"]]},

{"id":"c8911fad.47f72","type":"smooth","z":"b308f371.c9fa","name":"" ,"property":"payload","action":"mean","count":"25","round":"0","mult":"single", "x":780,"y":360,"wires":[["6270e4ac.2138fc","64858a7e.ecae14"]]}]

You can copy and paste this code block to import an ‘Absolute Amplitude’ function node, which takes the raw data from the Rshake Parser node and turns it into a more easily usable integer which describes the amount of absolute vibration over the past couple of seconds. After it there is a “smooth” node to take the mean of the first numbers in order for the data to be processed more easily. *Note: photos show “average magnitude” in place of “absolute amplitude”

To import code blocks like this, use the top right menu and select ‘import -> clipboard’ then a box will pop up allowing you to paste in the flow.

Your flow should now look similar to the photo above. The next step is to add the Dashboard “Gauge” Node, and attach it after the smooth node. Then, we have to configure the Gauge node. Make sure to make the range from zero to at least 50000 (remember not to put a comma in 50000). If you have not yet, you will also have to create a Dashboard group, by pressing the edit button next to the “group” field, and then pressing it again next to the “tab” field. Then click Update, then Done!

Now, we are ready to deploy!

Once deployed, open your local Node-Red Dashboard. You can access this page by entering http://yourIPaddress:1880/ui. For example: http://10.18.102.253:1880/ui

Now, bang the table! Jump up and down! How can you apply the ShakeMeter to something relevant in your home, school, or community?

So go get ‘Shakin!

Now go forward with your new skills and explore what projects you can do!

Keep Shaking!

Instructions

(PDF download)

LESSONS

EXERCISES

RESOURCES

Classroom Tips

Additional Links

GENERAL INFORMATION

- Products
- Shop
- Newsletter
- ShakeNet
- EQ View

COMPANY

- About Us
- Quality & Certifications
- Privacy Policy
- License
- Logo Files

GETTING STARTED

- Setup Guide
- Using the Data

POWERED BY RASPBERRY PI