MODULE: Real Time Geospatial Applications

LESSON: Node Red

Manfred Mittlboeck
Department of Geoinformatics - Z_GIS
University of Salzburg
Austria

GeoServices-4-Sustainalibity (GeoS4S)
ERASMUS+ Project No. 561716-EPP-1-2015-1-AT-EPPKA2-CBHE-JP Co-funded by the European Commission

Contents / Learning Objectives

- Why Node Red
- What is Node-Red
- Javascript – node.js – node-red
- Node-red 1st steps

Why Node-Red

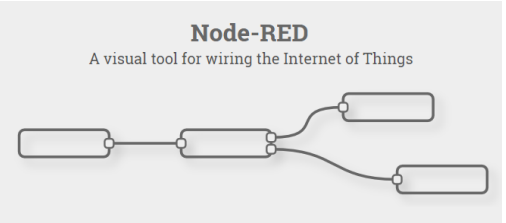
- The Internet of Things does not have a one-size-fits-all solution.
  - IoT often requires pulling together different device APIs and online services in new and interesting ways.
  - Time spent figuring out how to access a Serial port, or to complete API calls (e.g. an OAuth flow against Twitter is not an easy task without spending an amount of time
  - IoT needs easy to use tools to bring together the different streams of events.
  - Standards are great and available – but often rarely used

Node-Red and IoT

- Node-Red has been developed @IBM with focus focused on the Internet of Things for
  - connecting devices to processing and processing to devices
- Node-RED is an example of a flow-based programming model
  - messages representing events flow between nodes, triggering processing that results in output.
  - this flow-based programming model maps well to typical IoT applications which are characterised by real-world events that trigger some sort of processing which in turn results in real-world actions.

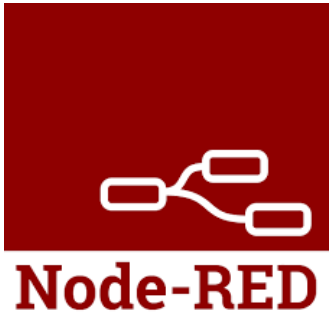
What is Node-Red

- Node-RED is a
  - The internet does not have a one-size-fits-all solution
  - powerful tool for building Internet of Things (IoT) applications with a focus **on simplifying the 'wiring together' of code blocks** to carry out tasks.
  - It **uses a visual programming approach** that allows developers to
    - connect predefined code blocks, known as 'nodes', together to perform a task.
  - www.nodered.org



What is Node-Red

- Node-RED is
  - A browser based application composition tool experience
  - A lightweight proof of concept runtime
  - Easy to use and to extend for simple tasks
  - A great way to try…
    - “can I just get this data from here to there?”
    - “and maybe change it just slightly along the way…”
  - It is not fully scalable and production ready



Node-Red Architecture

- Based on node.js v8-engine; so it's fast
  - powerful HTTP server that can be modified with the programmer 's wanted functionality.
  - A basic server is implemented in 5 line s of code.
- Event-driven, asynchronous io; it's all about the events
- Single-threaded event-queue; built for fairness
- Javascript front and back; only one language runtime to deal with
  - Built using express, d3, jquery and ws

ERASMUS+ Project No. 561716-EPP-1-2015-1-AT-EPPKA2-CBHE-JP Co-funded by the European Commission

Node-Red introduction

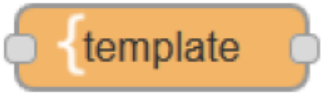
- Inject node

  - Allows manual triggering of flows
  - Can inject events at scheduled intervals

- Debug node

  - Show message content; either payload or entire object

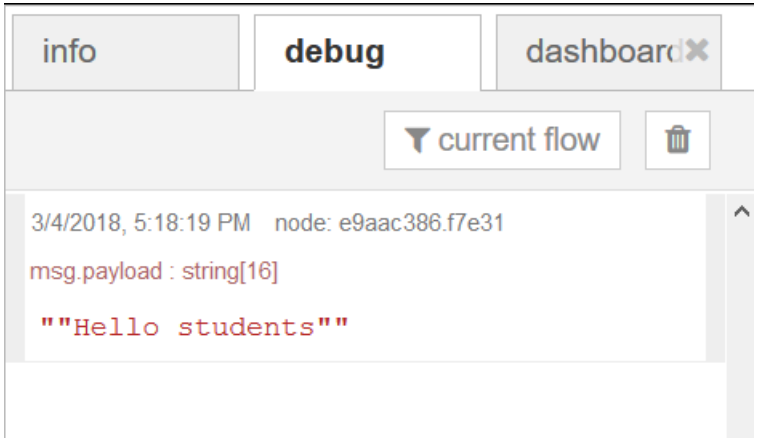
- Template Node



  - Modifies the output based on a Mustache Template

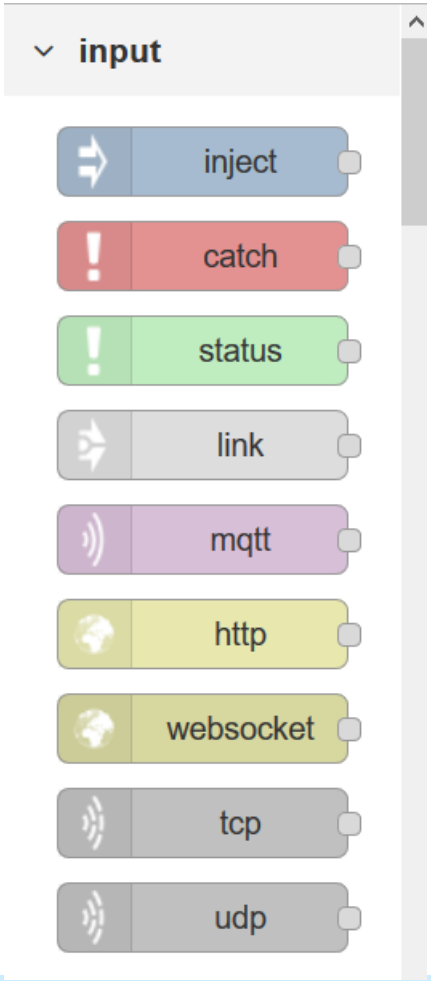
Node-Red 'Hello students'

- When you click on the Inject Node, it sends and event through the flow – triggering the template node and sending the result to the Debug node



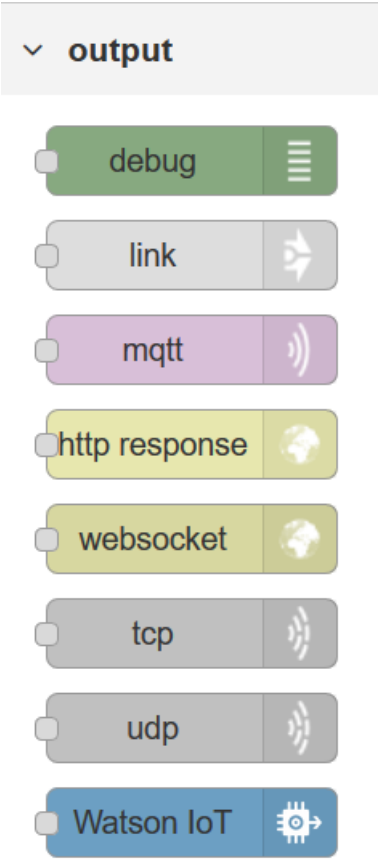
Node-Red - Input Nodes

- Inject
- HTTP – Act as an HTTP endpoint; great for building RESTful services
- Also can receive from
  - Websockets,
  - MQTT (pick your own broker),
  - IBM Watson IOT
  - TCP and UDP



Node-Red - Output Nodes

- HTTP Response; required as the final node when the input comes from an HTTP Request
  - Watson IoT – send events out to the attached IOT Foundation account
  - Twilio – send SMS messages via the Twilio service
  - Also can send requests through
    - TCP, UDP,
    - WebSockets.

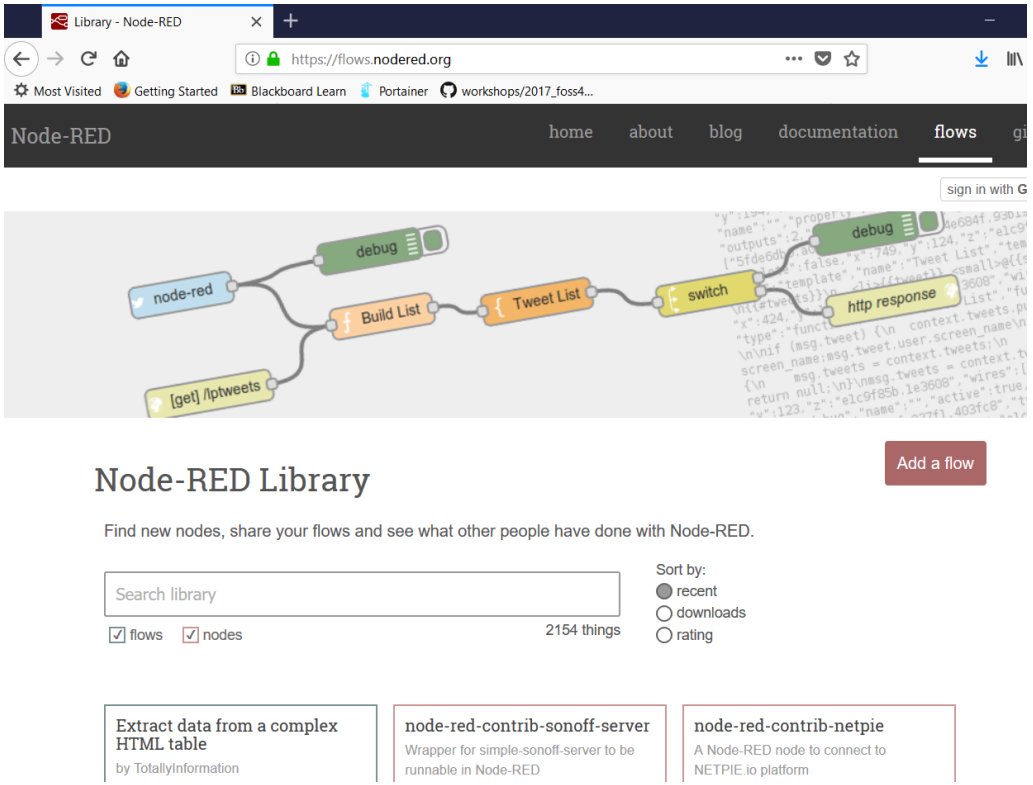


Node-Red – Function Node Types

- Function node
  - Returns messages
  - Run user-defined node.js code on the messages going by
  - Uses vm.createScript under the covers to sandbox execution
  - Console, util, Buffer etc. included
- Switch
  - Change flow to different options based on a comparison

Node-Red Online flow library

- Online flow library
  - Contributors add flows through Github



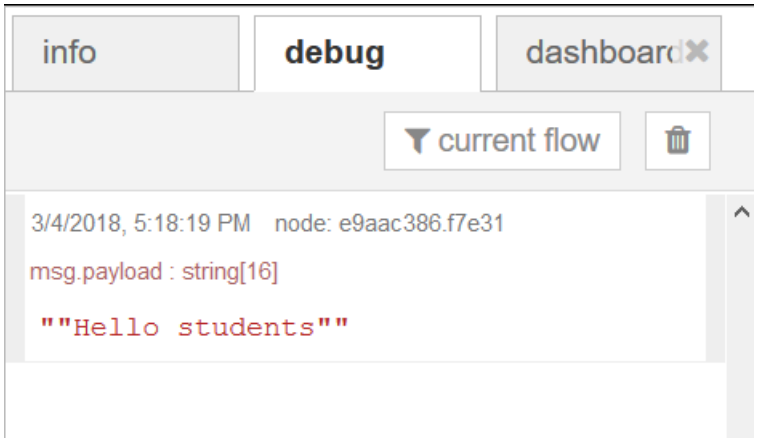
Node-Red Exercises Links

- A link summary is available @ https://geos4s.geo.sbg.ac.at
  - The used password is: Salzach2017$

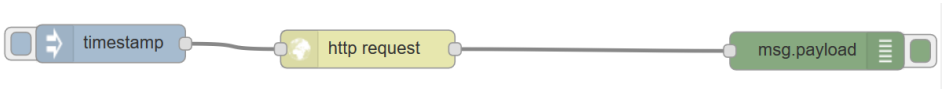
ERASMUS+ Project No. 561716-EPP-1-2015-1-AT-EPPKA2-CBHE-JP Co-funded by the European Commission

Node-Red Exercise 1

- Inject 1st string and output to debug console



Node-Red Exercise 2



- Inject 1st website XML and output debug console
  - METAR weather data to grab via https://aviationweather.gov
  - METAR Example Queries: https://aviationweather.gov/adds/dataserver/metars/MetarExamples.php
    - Example Salzburg (LOWS) - https://aviationweather.gov/adds/dataserver_current/httpparam?dataSource=metars&requestType=retrieve&format=xml&stationString=LOWS&hoursBeforeNow=0.5
    - Decode METAR Information http://weatherfaqs.org.uk/node/197
    - Find ICAO Code https://en.wikipedia.org/wiki/ICAO_airport_code#Prefixes

Node-Red Exercise 3

- METAR example XML response

This XML file does not appear to have any style information associated with it. The document tree is shown below.


```
▼<response xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XML-Schema-instance" version="1.2"
 xsi:noNamespaceSchemaLocation="http://aviationweather.gov/adds/schema/metar1_2.xsd">
   <request_index>72707379</request_index>
   <data_source name="metars"/>
   <request type="retrieve"/>
   <errors/>
   <warnings/>
   <time_taken_ms>5</time_taken_ms>
  ▼<data num_results="1">
    ▼<METAR>
      ▼<raw_text>
          LOWS 050420Z AUTO 11003KT 070V150 9999 FEW140 BKN150 02/M02 Q0999
       </raw_text>
       <station_id>LOWS</station_id>
       <observation_time>2018-03-05T04:20:00Z</observation_time>
       <latitude>47.78</latitude>
       <longitude>13.02</longitude>
       <temp_c>2.0</temp_c>
       <dewpoint_c>-2.0</dewpoint_c>
       <wind_dir_degrees>110</wind_dir_degrees>
       <wind_speed_kt>3</wind_speed_kt>
       <visibility_statute_mi>6.21</visibility_statute_mi>
       <altim_in_hg>29.498032</altim_in_hg>
      ▼<quality_control_flags>
         <auto>TRUE</auto>
       </quality_control_flags>
       <sky_condition sky_cover="FEW" cloud_base_ft_agl="14000"/>
       <sky_condition sky_cover="BKN" cloud_base_ft_agl="15000"/>
       <flight_category>VFR</flight_category>
       <metar_type>METAR</metar_type>
       <elevation_m>430.0</elevation_m>
     </METAR>
   </data>
 </response>
```

Node-Red Exercise 3

- Extract METAR phenomenon's values

```
if(msg.payload.response.data[0]){
    var temperature = msg.payload.response.data[0].METAR[0].temp_c[0];
    var dewpoint = msg.payload.response.data[0].METAR[0].dewpoint_c[0];
    var windspeed = Math.round(msg.payload.response.data[0].METAR[0].wind_speed_kt[0] * 1.852); // Umrechnung von Knoten in km/h
    var winddirection = msg.payload.response.data[0].METAR[0].wind_dir_degrees[0];

    return [{ payload: temperature}, { payload: dewpoint}, { payload: windspeed}, { payload: winddirection}];
}
```

References

- Partners in ERASMUS+ Project ‘GeoServices-4-Sustainability‘

- Please see full list of references in the notes section