Building a Dashboard Using Fleet Edge Assurance APIs

**INMARSAT** > Product Group > Fleet Edge

A guide to building a dashboard in Grafana, using the Fleet Edge assurance REST APIs

# Introduction

Inmarsat provides an assurance REST API which allows access to assurance metrics from Fleet Edge devices. This documentation shows how the API can be used to build a dashboard in Grafana.

Grafana is an open source platform for creating interactive dashboards, with a wide range of data visualisations. For more information, and to install Grafana, visit [grafana.com/](https://grafana.com/).

## Architecture

To enable Grafana to gather data from the Inmarsat REST APIs, a bridge is needed. The Simple JSON data source in Grafana coupled with a Flask webserver, is provided as an example bridge. The Flask webserver exposes endpoints which query the Inmarsat API and return assurance data a specified JSON format. The Simple JSON data source then queries the Flask webserver’s endpoints, and feeds the data into Grafana.

Bridge

# Instructions

## Prerequisites

* Grafana installed. The installation instructions for Grafana can be found at <https://grafana.com/docs/>. The instructions in this document will differ if Grafana is installed as a Docker container.
* Python 3 installed, with the following libraries
  + Flask
  + Flask\_cors
  + Json
  + Pandas
  + Numpy
  + Datetime
  + Requests

## Install the Simple Json plugin

* Run Grafana’s command line interface tool.

In windows, this can be found in:

<Your Installation Folder>\grafana-6.0.2.windows-amd64\grafana-6.0.2\bin\grafana-cli.exe

* Install the Simple Json plugin using the command:

grafana-cli plugins install grafana-simple-json-datasource

* Restart Grafana.

For more information, see <https://grafana.com/plugins/grafana-simple-json-datasource/installation> .

## Run the Python bridge

* In your console, run the command

Python flask\_bridge.py

The python bridge will be running at <http://localhost:5001/>.

## Configure the data source in Grafana

* Open Grafana
* Select the ‘Data Sources’ tab on the ‘Configuration’ page
* Add a new data source
* Select the Simple Json data source type
* In the URL box, enter <http://localhost:5001/>
* Check the boxes ‘Basic Auth’ and ‘With Credentials’
  + You may leave the ‘User’ box blank
  + In the ‘Password’ box enter your Inmarsat API key
* Select ‘Save & Test’

Grafana should report ‘Data source is working’.

## Creating a dashboard

* Select ‘New Dashboard’, from the sidebar

To select edge IDs:

* Select ‘Dashboard Settings’
* Under the ‘Variables’ tab, add a new variable
  + Name the variable however you would like
  + Select ‘Ad hoc filters’, under ‘type’
  + Select the Simple JSON data source, under ‘options’
* Returning to your dashboard, create instances of the new variable for each Fleet Edge device ID.

If this variable is not assigned a correct Fleet Edge ID, the dashboard will not display any data.

## Adding a map to the dashboard

To add a map, you will need to install the WorldMap Panel plugin.

* In Grafana’s command line interface tool, run the command

grafana-cli plugins install grafana-worldmap-panel

* Restart Grafana
* In your dashboard, create a new panel
* Selecting the query type ‘table’, select the metric ‘FBBTerminal\_Latitude’ or ‘IDirectCoreModule\_Latitude’
* Add a new query, of type ‘table’, for the metric ‘FBBTerminal\_Longitude’ or ‘IDirectCoreModule\_Longitude’
* Under the ‘Visualization’ tab
  + Select ‘Worldmap Panel’
  + Under ‘Location Data’, select ‘table’
  + Under ‘Aggregation’, select ‘current’
  + Under ‘Table Query Format’, select ‘co-ordinates’
  + Under ‘Location Name Field’ enter ‘device\_id’.
  + Under ‘Metric Field’, enter ‘time
  + Under ‘Latitude Field’, enter ‘FBBTerminal\_Latitude’
  + Under ‘Longitude Field’, enter‘FBBTerminal\_Longitude’
  + Change the min and max circle sizes, to suite. Making the max circle size larger than the min circle size, will show older data points as smaller circles.

For more information about the WorldMap Panel, see <https://grafana.com/plugins/grafana-worldmap-panel>.