openSenseMap

Dokumentation

Table of Contents

openSenseMap

The openSenseMap (OSeM) is a webplatform which provides upload, visualisation and analysis of location-specific sensordata.

Stations may be registered on the platform, which host one or more Sensors on a specific location. Data up- & download is done via the restful API[1].

Features

- timeseries visualization for each phenomenon
- filtering by various parameters
- spatial interpolation
- data download with bounding box

All sensor data is available for download under the Public Domain Dedication and License 1.0[1].

openSenseMap and it's API is open source software. Sourcecode and issuetracker are located here:

- openSenseMap[2]
- openSenseMap API[3]

> [1]. See 2.1 REST API ↩

> [1]. http://opendatacommons.org/licenses/pddl/summary/ ↩

> [2]. https://github.com/sensebox/OpenSenseMap ↩

> [3]. https://github.com/sensebox/OpenSenseMap-API ↩

Registration on the OSeM

Register a Luftdaten.info Particulate Matter Sensor on the openSenseMap

You can send the data of your particulate matter sensor to the openSenseMap if you follow these steps.

1. Look up your sensor configuration and register[1] a new senseBox

- Using the web interface of your device, you can find out which sensors are attached (Fig. 1[1]).
- Go to https://opensensemap.org/register[2], fill out your name, location and exposure.
- In the section Hardware select luftdaten.info. Now select the correct sensor configuration matching your local sensor configuration (Fig. 2[2]).
- Finish the registration.
- Attention: Copy your senseBox ID. This is a 24 character long string looking like this: 58a88c6b650831d8a3625e01
- If you registered with a correct mail address, the senseBox ID will also be sent via mail.

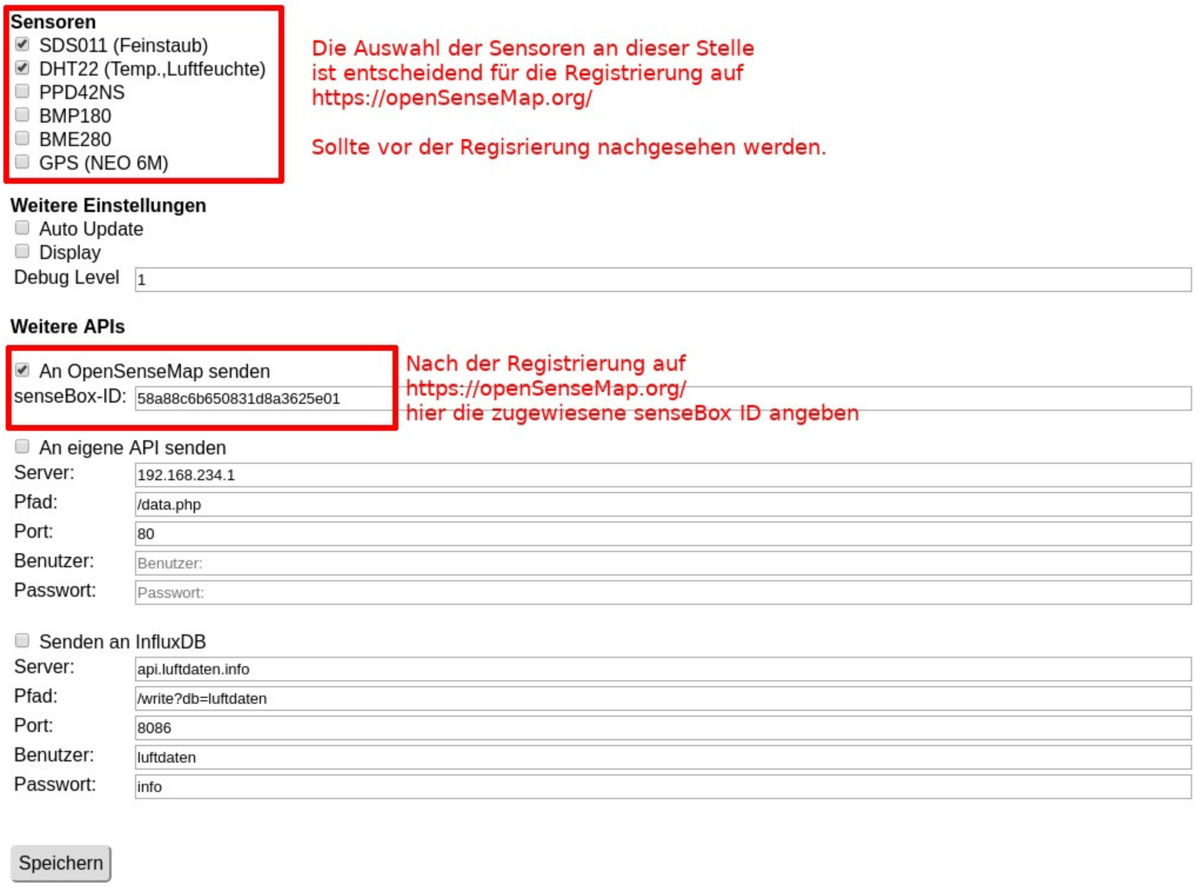
2. Configure your device

- Go to the web interface of your device
- Open the configuration page
- Paste your senseBox ID in the field next to Send to openSenseMap and check the box.
- Save the configuration with the button on the bottom of the page.

Done

Your device should now send its data to the openSenseMap!

Figure 1: Webinterface particulate matter sensor



Sensoren

☑ SDS011 (Feinstaub)
☑ DHT22 (Temp.,Luftfeuchte)
☐ PPD42NS
☐ BMP180
☐ BME280
☐ GPS (NEO 6M)

Die Auswahl der Sensoren an dieser Stelle ist entscheidend für die Registrierung auf https://openSenseMap.org/

Sollte vor der Regisrierung nachgesehen werden.

Weitere Einstellungen

☐ Auto Update
☐ Display
Debug Level: 1

Weitere APIs

☑ An OpenSenseMap senden
senseBox-ID: 58a88c6b650831d8a3625e01

Nach der Registrierung auf https://openSenseMap.org/ hier die zugewiesene senseBox ID angeben

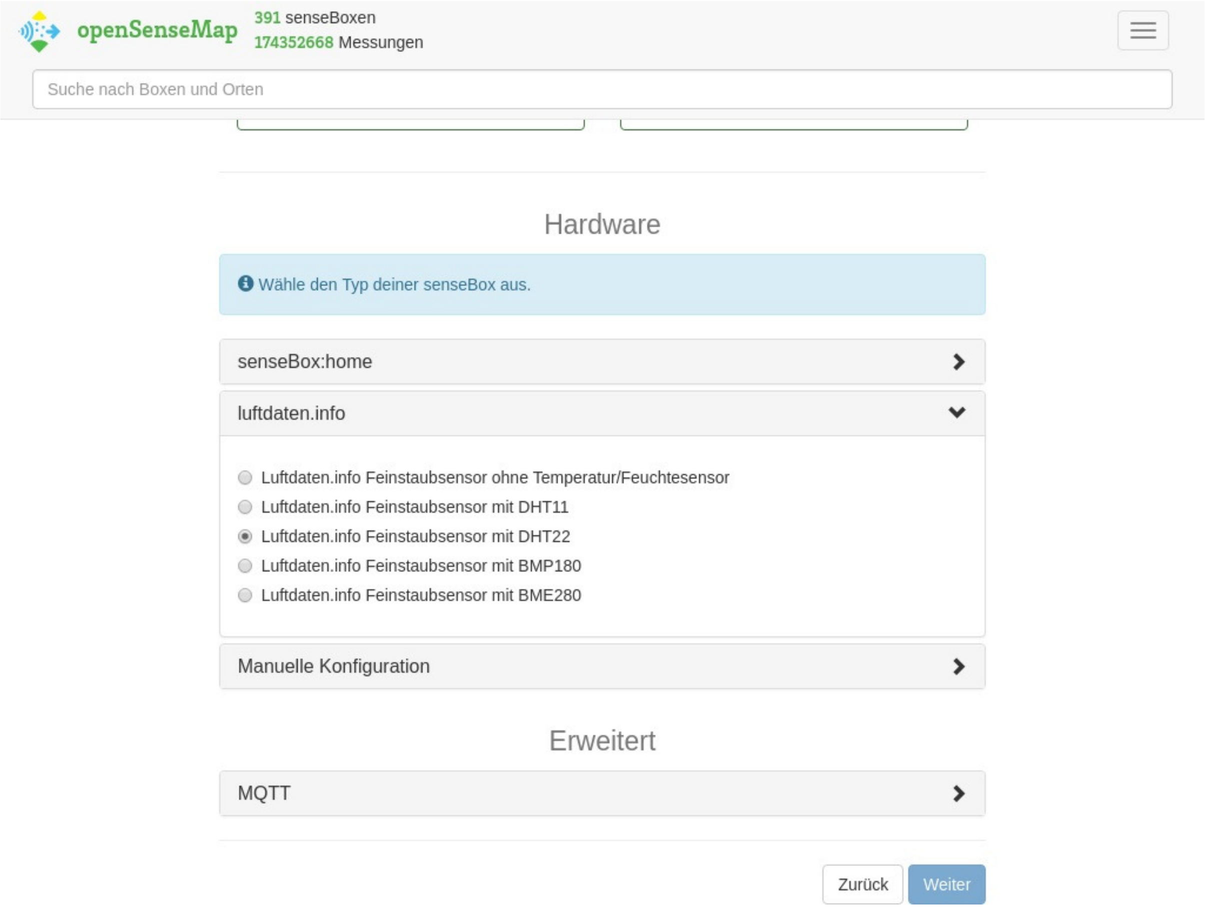
☐ An eigene API senden

Server:	192.168.234.1
Pfad:	/data.php
Port:	80
Benutzer:	Benutzer:
Passwort:	Passwort:

☐ Senden an InfluxDB

Server:	api.luftdaten.info
Pfad:	/write?db=luftdaten
Port:	8086
Benutzer:	luftdaten
Passwort:	info

Figure 2: Registration on openSenseMap



Hardware

Wähle den Typ deiner senseBox aus.

senseBox:home

luftdaten.info

○ Luftdaten.info Feinstaubsensor ohne Temperatur/Feuchtesensor
○ Luftdaten.info Feinstaubsensor mit DHT11
● Luftdaten.info Feinstaubsensor mit DHT22
○ Luftdaten.info Feinstaubsensor mit BMP180
○ Luftdaten.info Feinstaubsensor mit BME280

Manuelle Konfiguration

Erweitert

MQTT

[1]. https://opensensemap.org/register ↩

[1]. See 1.2.1 Luftdaten.info Airrohr > figure-1-webinterface-particualte-matter-sensor ↩

[2]. https://opensensemap.org/register ↩

[2]. See 1.2.1 Luftdaten.info Airrohr > figurre-2-registration-on-opensensemap ↩

Submitting data through MQTT

The openSenseMap is able to receive measurements through its internal MQTT client. There is no openSenseMap MQTT broker, connections are made with a 13 character long client id with `osem_` as prefix.

Connection settings must be configured per senseBox.

The following settings can be made:

URL

The address of the MQTT broker. Should look like this: `mqtt://username:password@hostname.of.mqtt.broker`

Topic

The MQTT topic. Example: `home/temperatures/outside`

Messageformat

Either `json` or `csv` . Formats are documented here[1].

Decode Options

A JSON object. Allows to specify a `jsonPath` expression to specify the position of the json encoded message. Example: `{"jsonPath":"$.payload_fields"}`

Connection Options

A JSON object. Allows to configure the mqtt client. Keys `keepAlive` , `reschedulePings` , `clientId` , `username` and `password` of https://github.com/mqttjs/MQTT.js#client[2] are allowed.

> [1]. https://docs.opensensemap.org/#api-Measurements-postNewMeasurements ↩

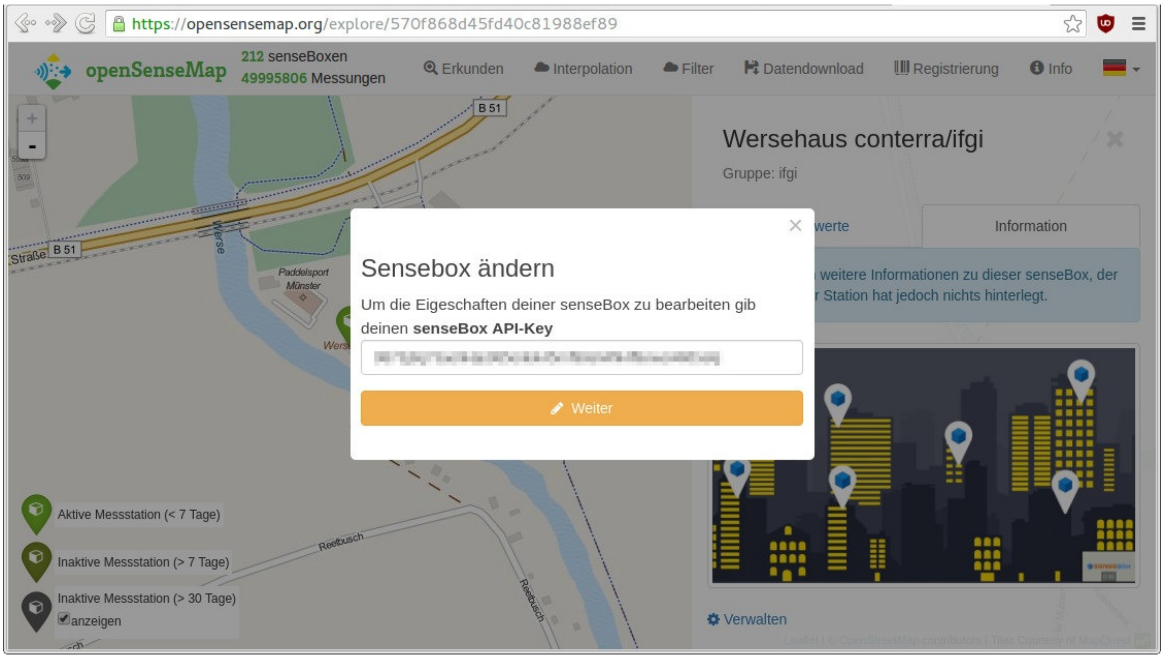
> [2]. https://github.com/mqttjs/MQTT.js#client ↩

Modifying a station

All properties of a station may be changed after the registration.

To do this, authorization with the API-key is required, which was sent to you in the registration e-mail!

1. Select your station on the map by clicking on the marker on the map.
2. Select the tab "Info" in the sidebar and click "Manage".
3. Enter your API-key in the dialog.



Sensebox ändern

Um die Eigeschaften deiner senseBox zu bearbeiten gib deinen **senseBox API-Key**

[redacted]

Weiter

https://opensensemap.org/explore/570f868d45fd40c81988ef89
openSenseMap
212 senseBoxen
49995806 Messungen
Erkunden
Interpolation
Filter
Datendownload
Registrierung
Info
Wersehaus conterra/ifgi
Gruppe: ifgi
werte
Information
weitere Informationen zu dieser senseBox, der
Station hat jedoch nichts hinterlegt.
Verwalten
B 51
Straße
Werse
Paddelsport Münster
Reetbusch
Aktive Messstation (< 7 Tage)
Inaktive Messstation (> 7 Tage)
Inaktive Messstation (> 30 Tage)
☑anzeigen

4. Make your desired changes in the appearing form. You may edit metadata, geolocation, photo, as well as the stations sensor configuration.

   > Hint: If you have added a new sensor and want to download the updated arduino-sketch, a page-reload after saving is required.

5. Click "save" or "cancel" in the top of the dialog to apply or discard your changes.

Deleting a station

Follow the steps under "Modifying a station"[1], then type `DELETE` in the textfield "Delete senseBox" and confirm.

> warning: All associated sensor data will be permanently deleted!

> [1]. See 1.3 Editing a station > modifying-a-station ↩

Data download

Data analysis

Filter

Interpolation

openSenseMap RESTful API

The openSenseMap provides a REST API, which can be used to query & post senseBox metadata and measurements. The endpoint is https://api.opensensemap.org/.

The API documentation can be found here[1].

[1] https://docs.opensensemap.org ↩

Applications using the openSenseMap API

There already are a couple of tools and integrations for the openSenseMap that made by the community.

> If you know about an integration that is not listed here, let us know!

Visualization

- DevelopmentSeed Dashboard[1]: great looking dashboard for measurements of a senseBox
- HPI Makerclub Dashboard[2] (demo[3])
- senseBox Dashboard[4]: shows current sensor values of a senseBox
- senseBox Widget[1]

Data Analysis

- `opensensmapr` R client[7]: package for analysis of observations within the R statistical environment

Sensor Firmware

- Raspi cloud coverage sensor[8]
- GSM water sensor[9]
- mobile GPS senseBox[10]
- BigGIS LoRa senseBox[11]

Misc

- Alexa senseBox-Skill[12]
- openhab2 smart home senseBox Integration[13]
- openSenseMap Game[14]: score points for guessing the location of current weather conditions correctly!

> [1]. https://github.com/developmentseed/sense ↩

> [2]. https://github.com/HPIMakerKlub/sensebox ↩

> [3]. http://rawgit.com/HPIMakerKlub/sensebox/master/statistics/sensor.html?senseBoxID=5719c4037514d05c121e317c ↩

> [4]. https://github.com/sensebox/sensebox-dashboard ↩

> [1]. See 2.3 HTML Widget ↩

> [5]. https://github.com/Avipsa1/Sensebox ↩

> [6]. https://github.com/sensebox/Innotruck ↩

> [7]. https://github.com/noerw/opensensmapR ↩

> [8]. https://github.com/felixerdy/senseBox-cloud ↩

> [9]. https://github.com/felixerdy/GSM-Temperature-senseBox ↩

> [10]. https://github.com/noerw/mobile-sensebox ↩

> [11]. https://github.com/biggis-project/sensebox-station ↩

> [12]. https://github.com/Zeygon/alexa-sensebox ↩

> [13]. https://github.com/hakan42/openhab2-addons/tree/sensebox-binding/addons/binding/org.openhab.binding.sensebox ↩

> [14]. https://github.com/MaxMoody/Senseboxgamification ↩

openSenseMap Widget

To display the data of a senseBox not only on `opensensemap.org`, but for example also on your own website, you may use our widget.

Example

Usage

To include the widget, you just need to add an `iframe` to your page.

Step 1: Find your senseBox ID

Go to https://opensensemap.org/[1] and open a senseBox of your choice. Now copy the last portion of the address bar of your browser. This is your senseBox ID.

Step 2: Insert HTML into your page

In order to include the widget into your web page, just include the following html into your page at the desired location.

Replace the `YOUR-SENSEBOX-ID` in the `src` attribute with the senseBox ID from step 1.

```
<iframe
  src="https://sensebox.github.io/opensensemap-widget/iframe.html?senseboxId=YOUR-SENSEBOX-ID"
  width="400"
  height="600"
  marginwidth="8" marginheight="8"
  hspace="0" vspace="0"
  frameborder="0"
  scrolling="no"
></iframe>
```

You can play around with the `height` and `width` attributes.

You can find the source code of this widget on GitHub[2].

> [1]. https://opensensemap.org/ ↩

> [2]. https://github.com/sensebox/opensensemap-widget ↩