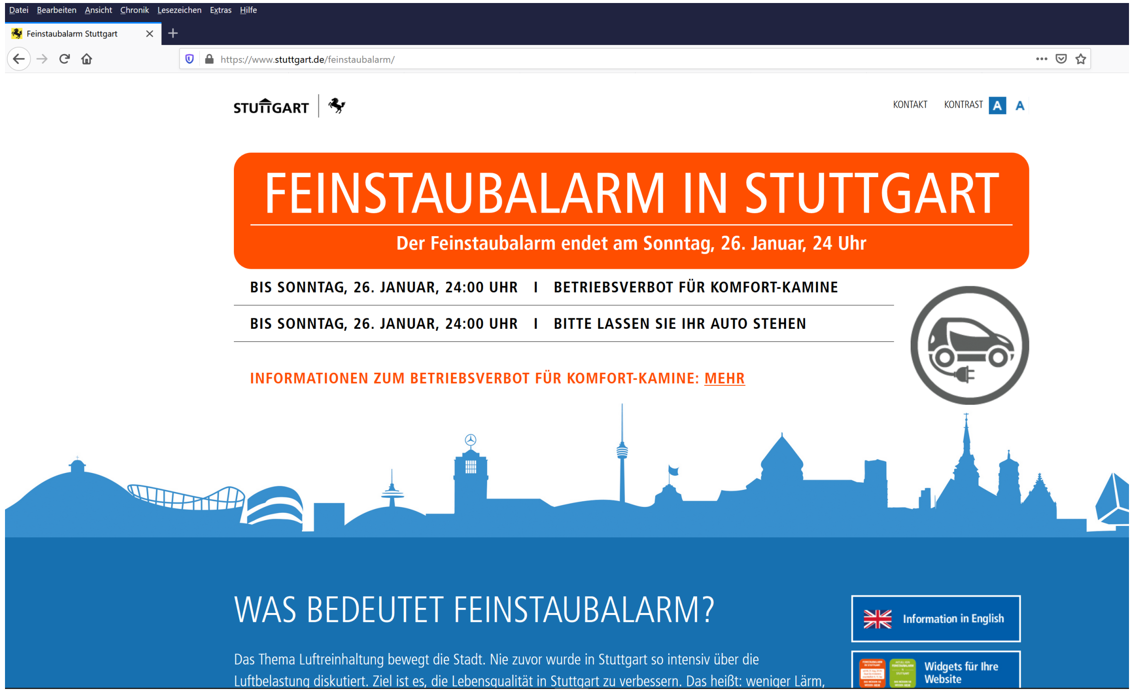
Analyzing air quality in Stuttgart using official and Citizen-Science data

SEMINAR: GIS ANALYSES WITH FOSSGIS
SPEAKERS: AMANDUS BUTZER, JULIAN KÄFLEIN
TEACHER: CHRISTINA LUDWIG
29.01.2019

Motivation



Source: https://www.stuttgart.de/feinstaubalarm/

ENVIRONMENT

Stuttgart: Germany's 'Beijing' for air pollution?

In German "car city" Stuttgart, air pollution has reached harmful levels - again. Authorities are trying "soft" appeals to the public - but environmentalists blame Germany's strong car lobby for the smoggy air.

https://www.dw.com/en/stuttgart-germanys-beijing-for-air-pollution/a-18991064

Motivation

Source: https://luftdaten.info/en/home-en/

Citizen Science – join in!

OK Lab Stuttgart is dedicated to the fine dust measurement with the Citizen Science project luftdaten.info. You and thousands of others around the world install self-built sensors on the outside their home. Luftdaten.info generates a continuously updated particular matter map from the transmitted data. Fine dust becomes visible.

Motivation

Particulate Matter (PM)

Defintion: Solid and liquid airborne particles under 10/2.5µm diameter (PM10/PM25)

Health hazards: Short term irritation/ inflammation of respiratory system, allergic reactions, asthma

Long-term increase of cardiovascular, respiratory diseases

Sources: Industrial processes, Heating, Traffic

<u>Stuttgart:</u>

Problem (1): Concentration of PM exceeding German/ European limits in recent

Reaction: Introduction of „Feinstaubalarm“ – plea to use public transport, ban of ‚comfort heating‘

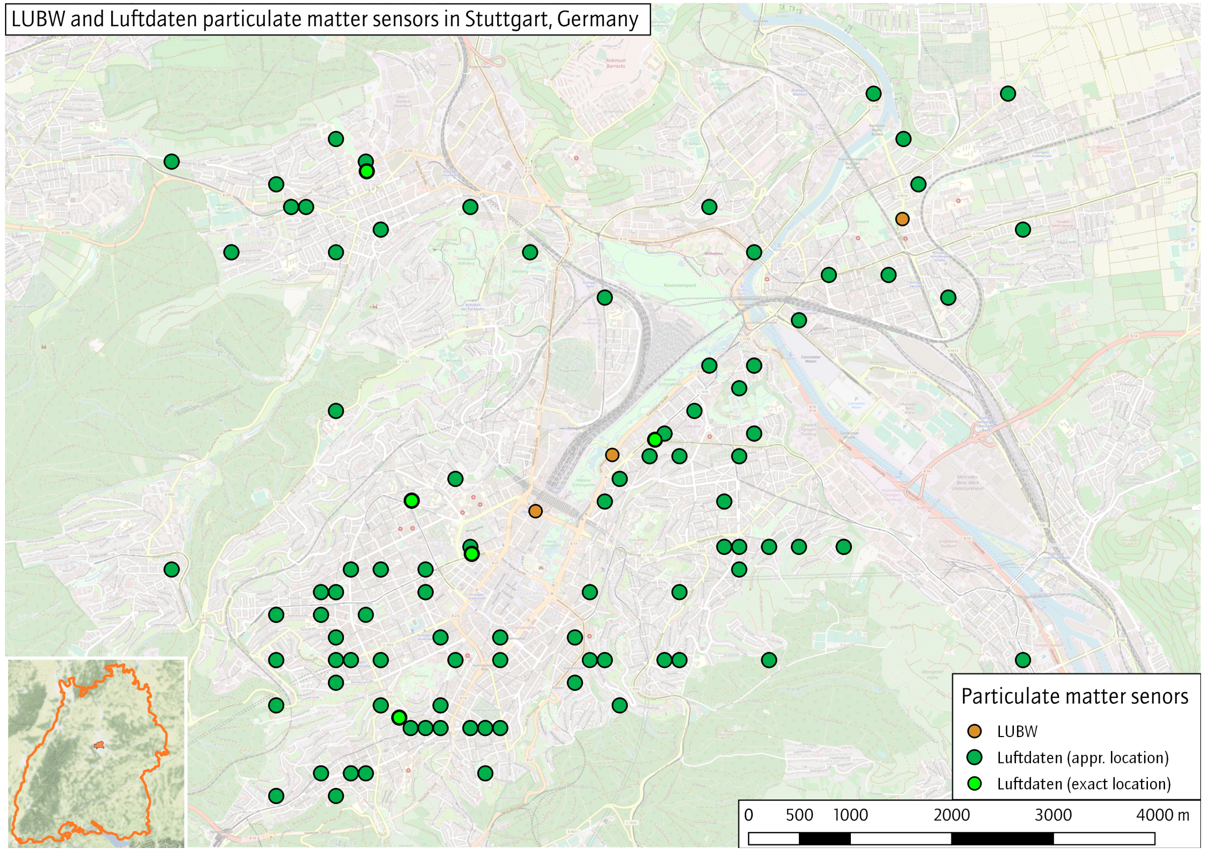
Problem (2): Amount of PM sensors – 3 sensors by Landesanstalt für Umwelt Baden-Württemberg (LUBW)

Scientific goal

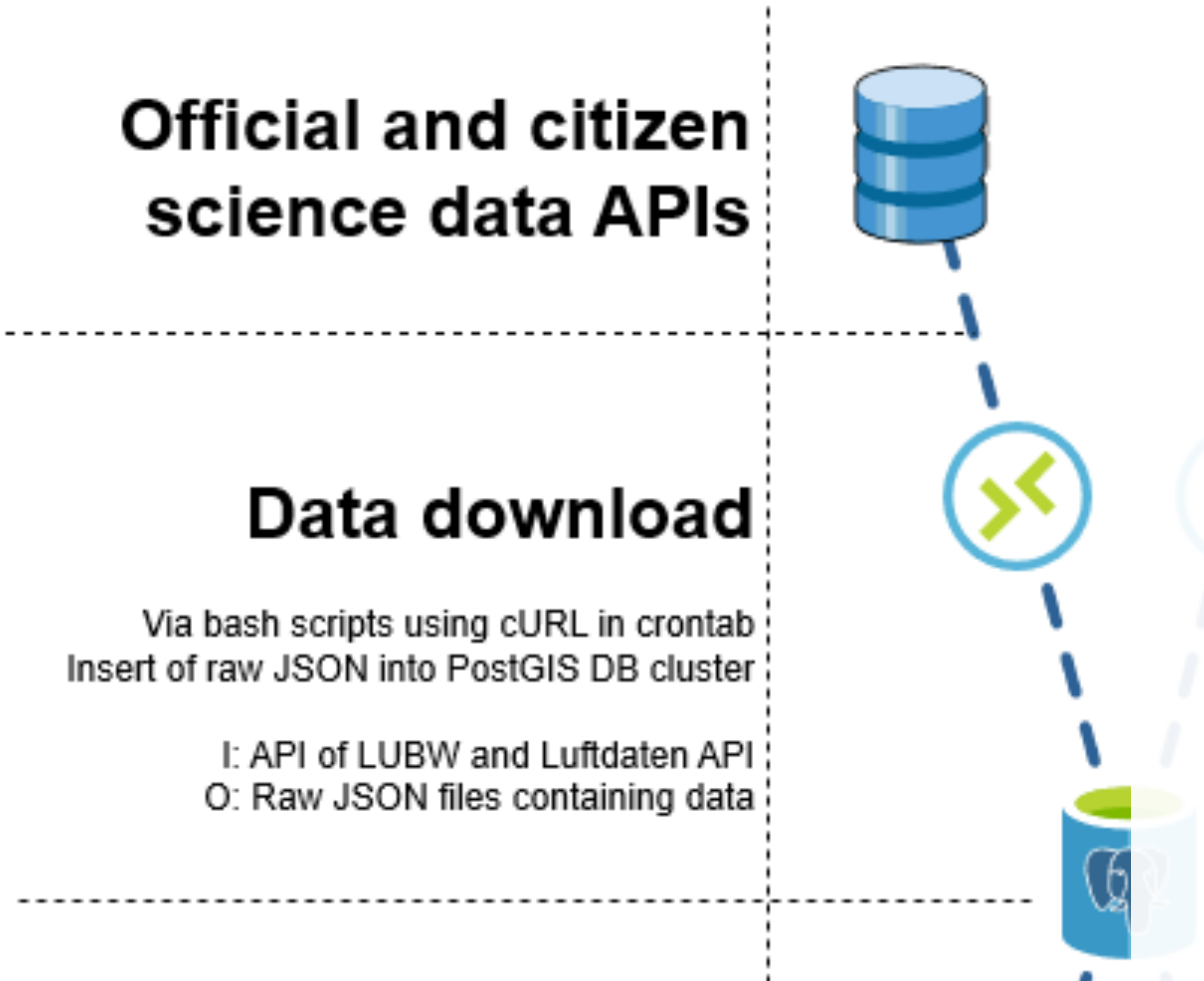
(How) can we analyze the particulate matter concentration

- in the city of Stuttgart
- based on Volunteered Geographic Information (VGI)
- using Free and Open Source Geographic Information Systems (FOSSGIS)
- in an automated process?

Research area



Workflow

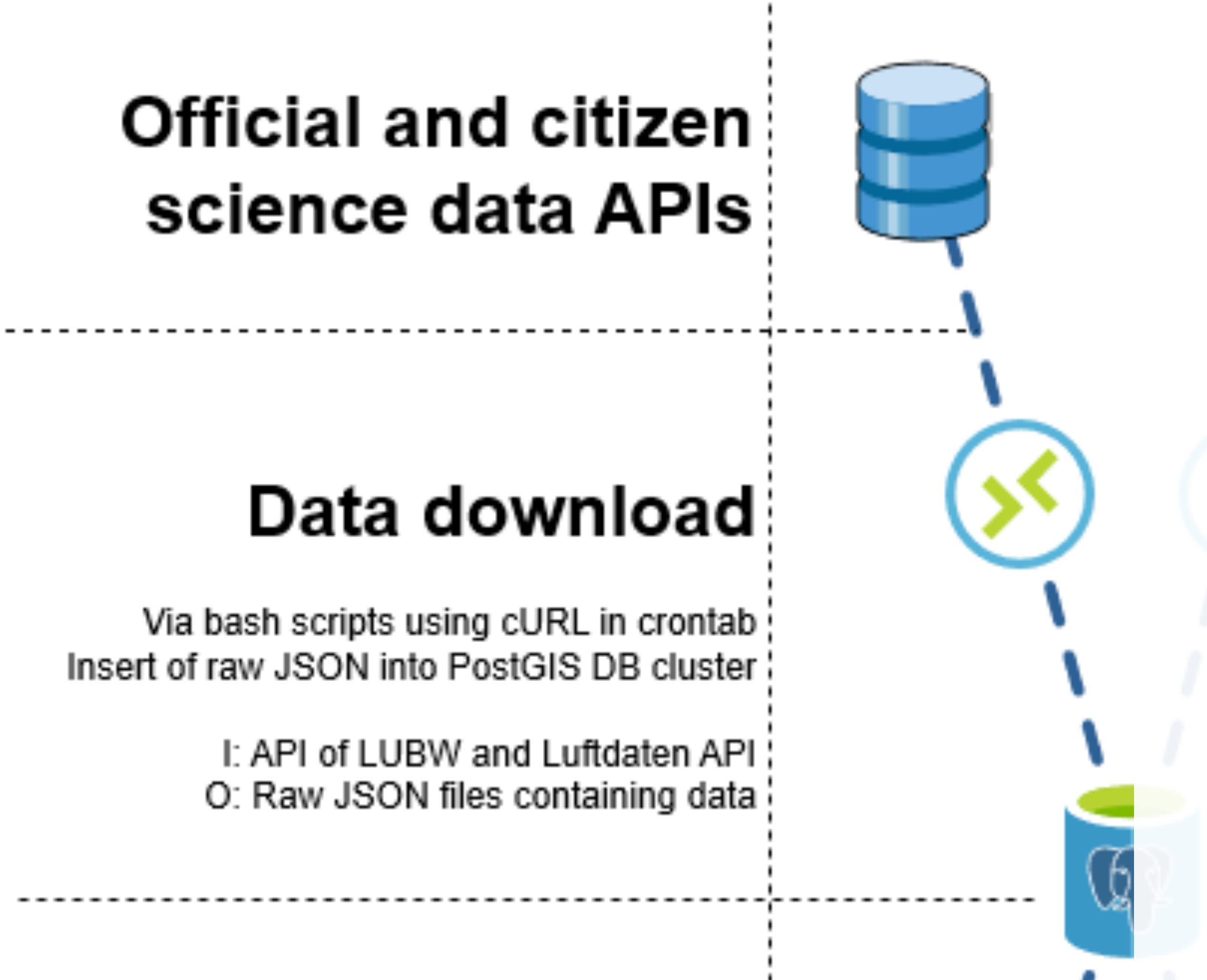



```bash
#!/bin/bash
while :
do
    OUTFILE=luftdaten_$(TZ=Europe/Berlin date +%Y%m%d_%H%M).geojson
    until curl --fail "https://data.sensor.community/static/v2/data.1h.json" \
    -o luftdaten_dl/$OUTFILE;
    echo "$OUTFILE heruntergeladen"
    do
        echo "$OUTFILE nicht heruntergeladen. Neuer Versuch in 30 Sekunden"
        sleep 30
    done

PGPASSWORD="postgres" psql -p 5430 -d luftdaten -U postgres /
-c "\copy input_raw_luftdaten FROM 'luftdaten_dl/$OUTFILE'"

PGPASSWORD="postgres" psql -p 5430 -d luftdaten -U postgres /
-c "SELECT daten.luftdaten_parse()"

sleep 15
done
```

Luftdaten:

1. Rate: Hourly values
2. Sensors: All (global)
3. Format: JSON
4. URL: https://data.sensor.community/static/v2/data.1h.json
5. Size: ~ 9MB (!)

Workflow

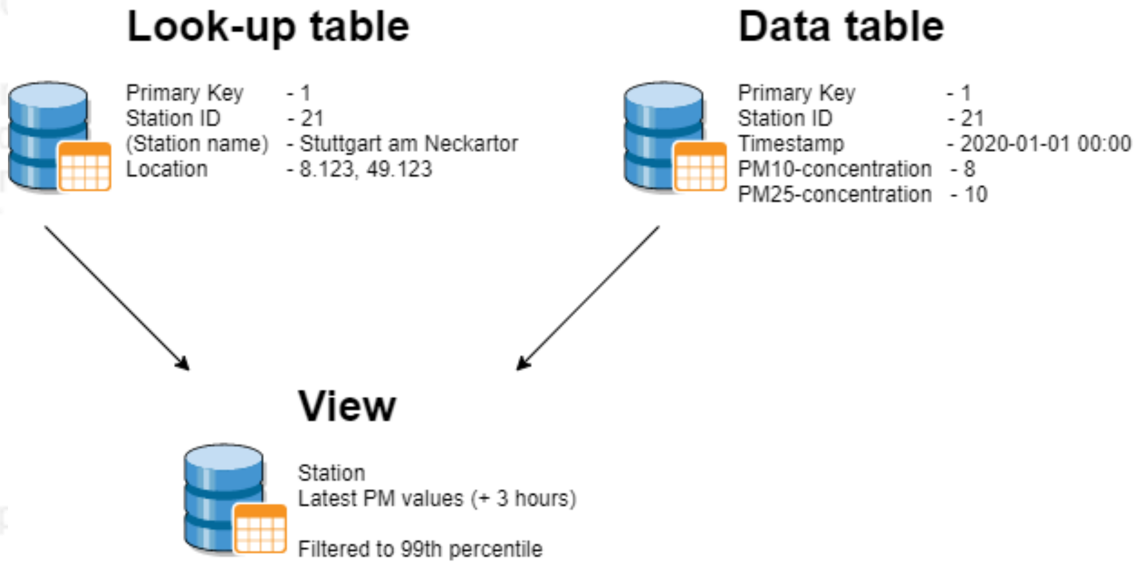



```bash
#!/bin/bash
while :
do
    OUTFILE=luftdaten_$(TZ=Europe/Berlin date +%Y%m%d_%H%M).geojson
    until curl --fail "https://data.sensor.community/static/v2/data.1h.json" \
    -o luftdaten_dl/$OUTFILE;
    echo "$OUTFILE heruntergeladen"
    do
        echo "$OUTFILE nicht heruntergeladen. Neuer Versuch in 30 Sekunden"
        sleep 30
    done

PGPASSWORD="postgres" psql -p 5430 -d luftdaten -U postgres /
-c "\copy input_raw_luftdaten FROM 'luftdaten_dl/$OUTFILE'"

PGPASSWORD="postgres" psql -p 5430 -d luftdaten -U postgres /
-c "SELECT daten.luftdaten_parse()"

sleep 15
done
```

LUBW

1. Rate: Hourly values
2. Sensor: All (Baden-Württemberg)
3. Format: GeoJSON
4. URL: https://lupo-messwerte.appspot.com/generic?table=bw_luft_stammdaten&limit=999&filter=aktiv:true;type:-Spot&order=NO2-today-latest-class
5. Size: 228KB

Workflow



1. Download data every hour
2. Insert raw data into staging database
3. Parse raw data:

   <u>3.1. LUBW:</u>

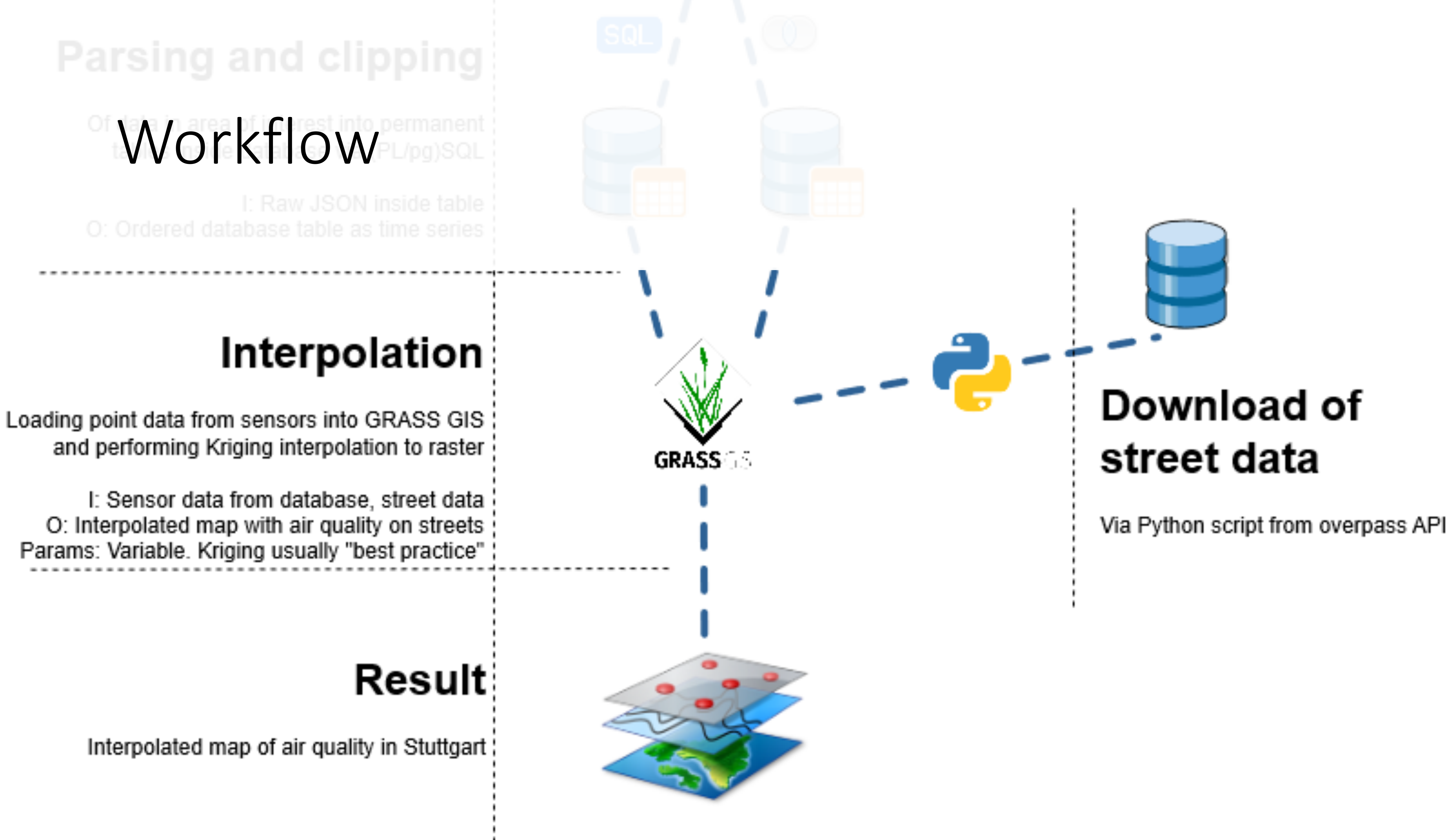
   Select all Stations within Stuttgart
   Select relevant air quality values

   <u>3.2 Luftdaten:</u>

   Build geometry from Lat/Lon
   Select only relevant values / sensors
   Clip stations within Stuttgart

4. Insert stations / values into tables

Workflow


Parsing and clipping
Of data in area of interest into permanent
SQL
I: Raw JSON inside table
O: Ordered database table as time series
Interpolation
Loading point data from sensors into GRASS GIS and performing Kriging interpolation to raster
I: Sensor data from database, street data
O: Interpolated map with air quality on streets
Params: Variable. Kriging usually "best practice"
GRASS
Download of street data
Via Python script from overpass API
Result
Interpolated map of air quality in Stuttgart

Analysis

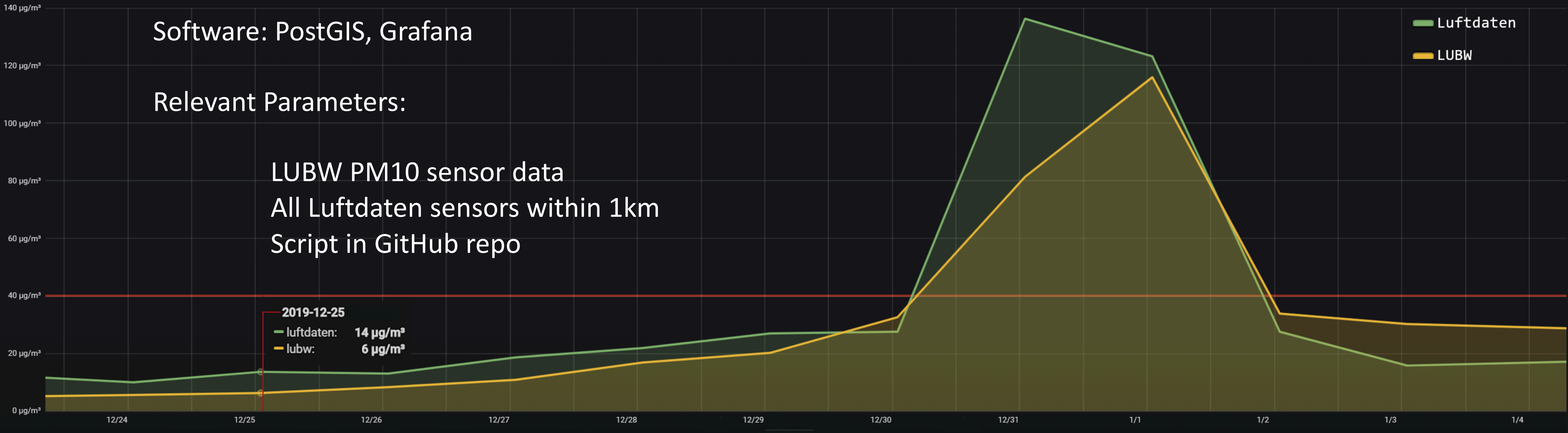
Daily average PM10 concentration for LUBW sensors vs Luftdaten sensors within 1000m distance

Method: Buffering, daily average comparison

Software: PostGIS, Grafana

Relevant Parameters:

- LUBW PM10 sensor data
- All Luftdaten sensors within 1km
- Script in GitHub repo



Analysis

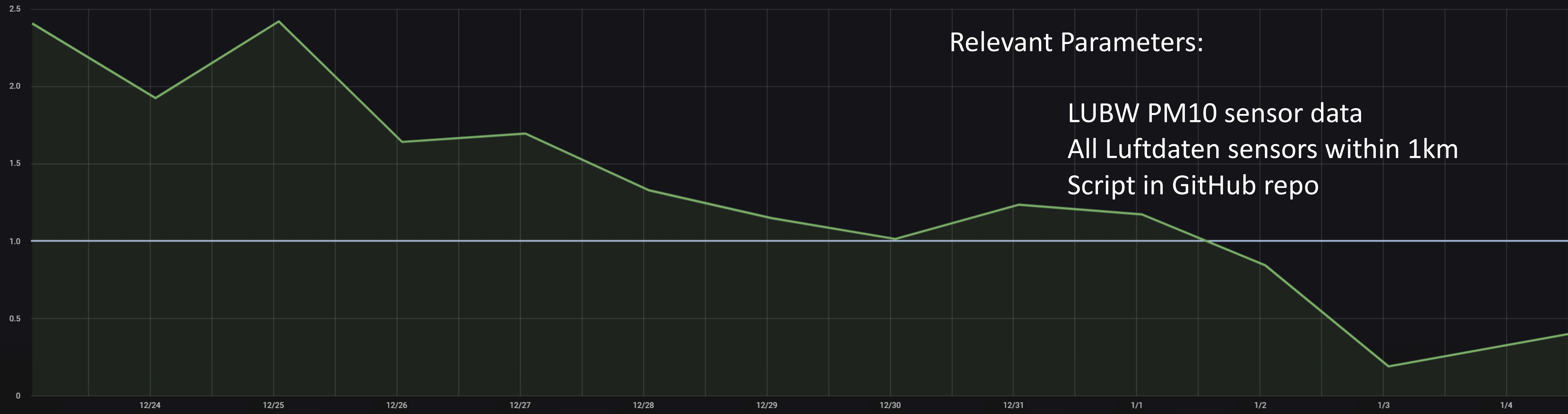
Daily average PM10 concentration – difference coefficient

Method: Buffering, daily coefficient comparison

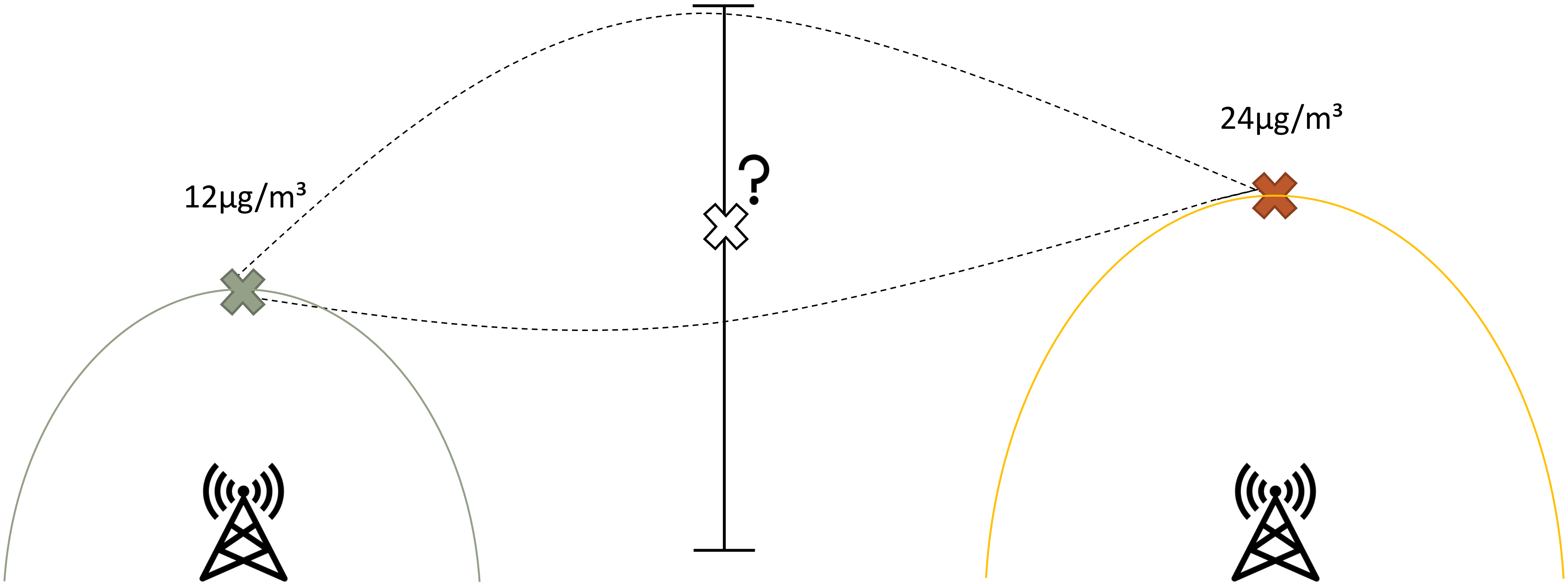
Software: PostGIS, Grafana

Relevant Parameters:

- LUBW PM10 sensor data
- All Luftdaten sensors within 1km
- Script in GitHub repo



Analysis - Interpolation



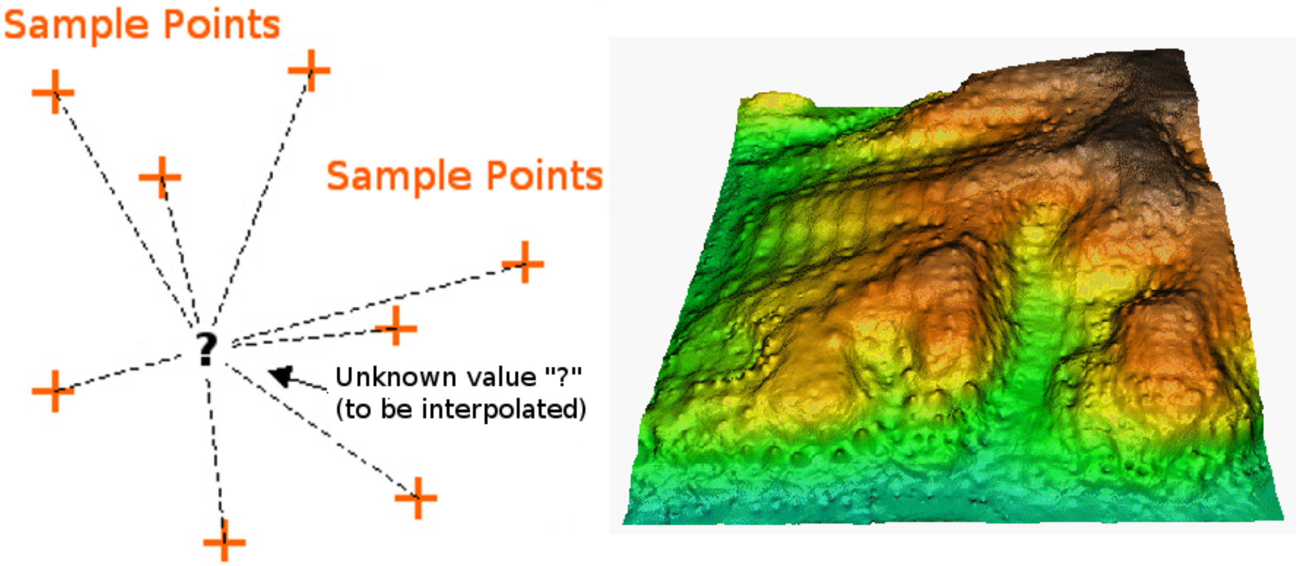
Method – Inverse Distance Weighted

Interpolated Point = linear weighted combination of nearby values

Power: Defines Distance Impact on value.

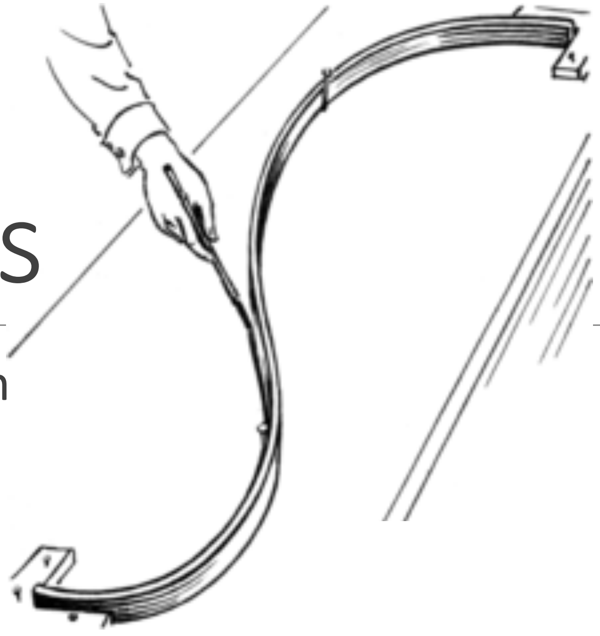
No interpolation outside value range

Good for data correlated to distance (noise)



https://www.battelleecology.org/sites/default/files/images/spatialData/idw_interpolation_QGIS.png

Method – B-Splines



https://upload.wikimedia.org/wikipedia/commons/thumb/f/fd/Spline_%28PSF%29.png/220px-Spline_%28PSF%29.png

Like IDW the surface passes exactly through datapoints

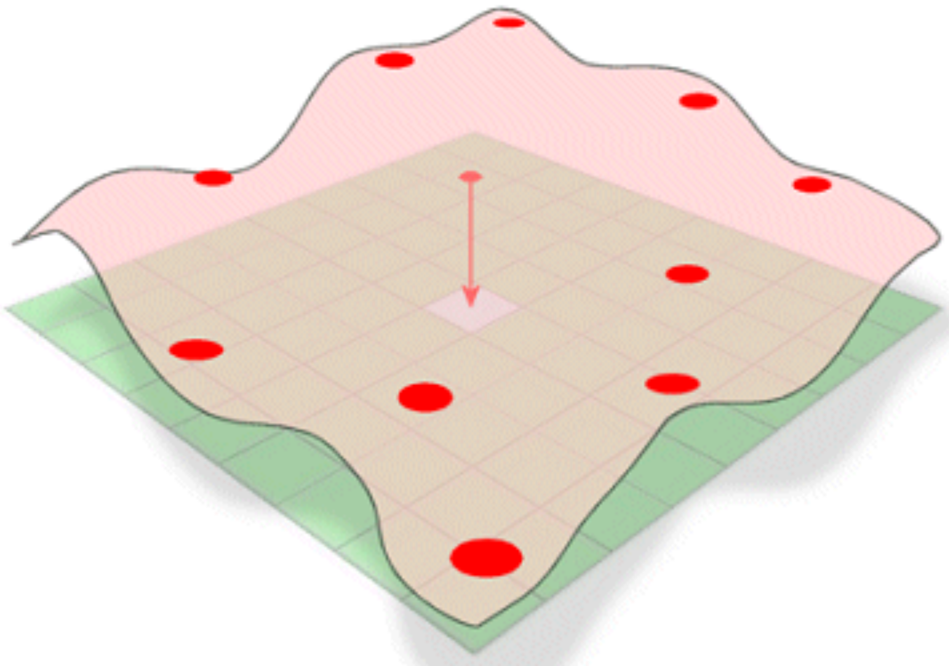
„Rubber Sheet glued to points“

Minimum curvature

Interpolation outside value range

Smoother surface

Using Tykhonov regularization



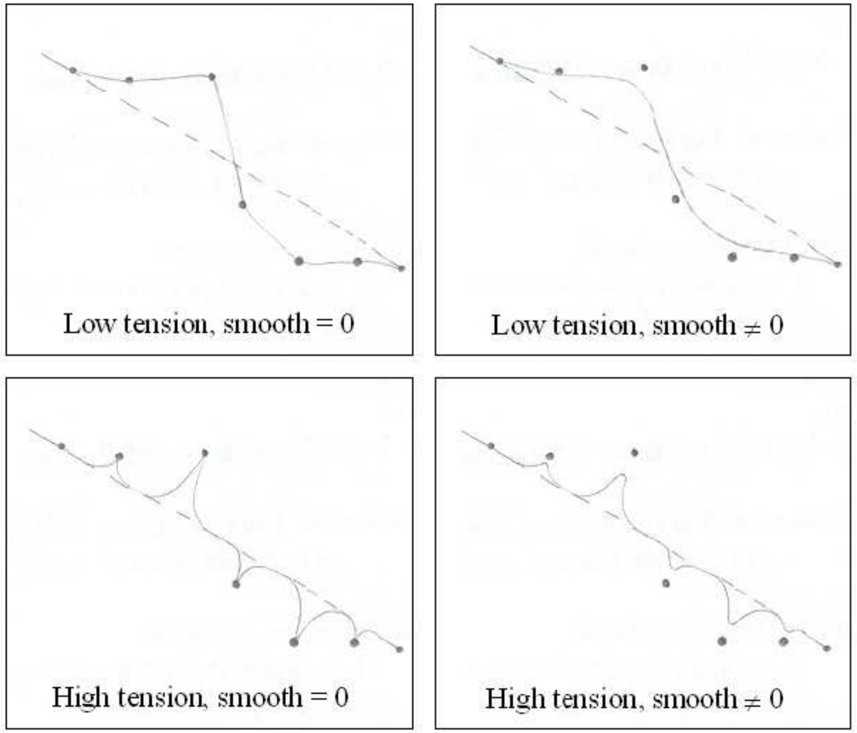
http://www.geography.hunter.cuny.edu/~jochen/GTECH361/lectures/lecture10/3Dconcepts/Spline_files/image001.gif

Method – Regularized Splines w. Tension

Tension and Smoothing of Splines possible

Plethora of output possible:
- elevation(basic value output)
- slope
- aspect
- curvatures
- treeseg

-> Best interpolation method so far



https://www.researchgate.net/profile/Domenico_Sguerso/publication/228701243/figure/fig12/AS:302005378666509@1449015101871/Sketch-of-RST-interpolation-results-with-high-low-tension-and-smoothing-equal-or-not-to.png

Method – Reclassification

EU-defined Common Air Quality Index (CAQI)

Using Background Index as station position not exact enough.

Method: r.reclass

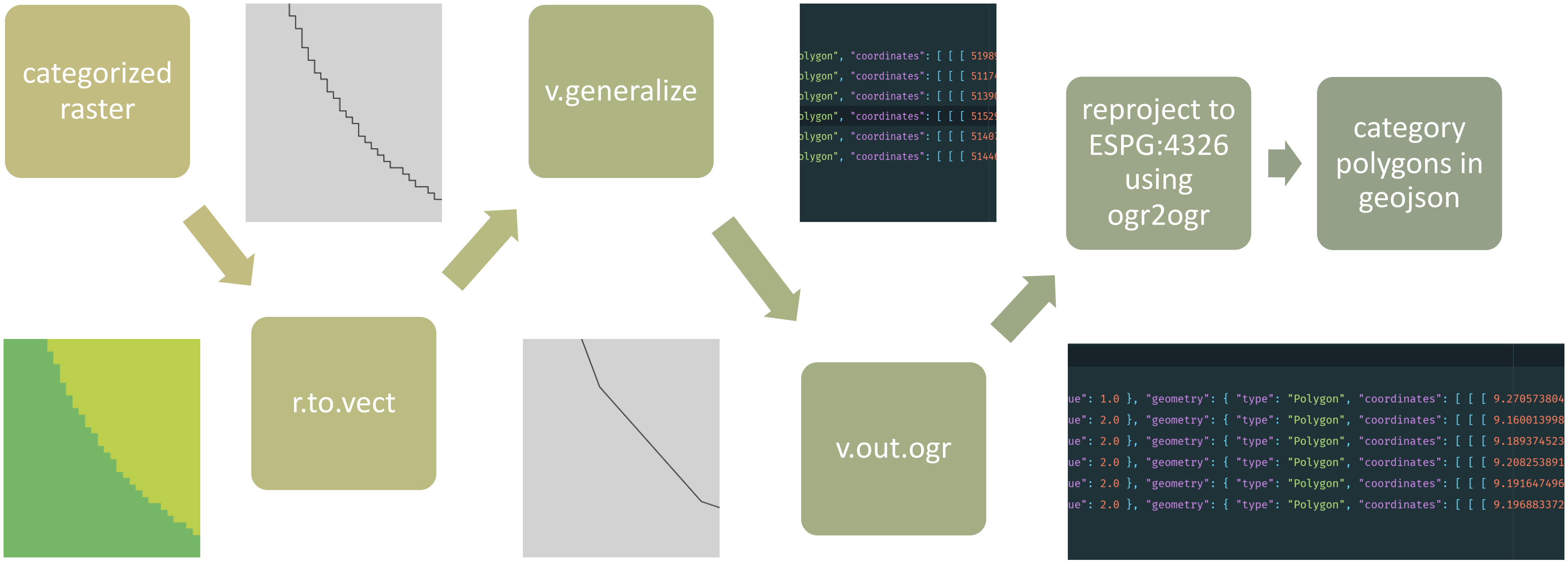
Common air quality index calculation grid

Index Class	Grid	ROADSIDE INDEX							BACKGROUND INDEX							
		Mandatory pollutant			Auxiliary pollutant				Mandatory pollutant				Auxiliary pollutant			
		NO2	PM10		PM2.5		CO		NO2	PM10		O3	PM2.5		CO	SO2
			1 hour	24 hours	1 hour	24 hours				1 hour	24 hours		1 hour	24 hours		
Very High	>100	>400	>180	>100	>110	>60	>20000		>400	>180	>100	>240	>110	>60	>20000	>500
High	100	400	180	100	110	60	20000		400	180	100	240	110	60	20000	500
	75	200	90	50	55	30	10000		200	90	50	180	55	30	10000	350
Medium	75	200	90	50	55	30	10000		200	90	50	180	55	30	10000	350
	50	100	50	30	30	20	7500		100	50	30	120	30	20	7500	100
Low	50	100	50	30	30	20	7500		100	50	30	120	30	20	7500	100
	25	50	25	15	15	10	5000		50	25	15	60	15	10	5000	50
Very Low	25	50	25	15	15	10	5000		50	25	15	60	15	10	5000	50
	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0

- NO2, O3, SO2: hourly value / maximum hourly value in µg/m3
- PM10, PM2.5: hourly value / maximum hourly value or adjusted daily average in µg/m3
- CO: 8 hours moving average / maximum 8 hours moving average in µg/m3

http://www.airqualitynow.eu/about_indices_definition.php

Method – Advanced Raster to GeoJson



Analysis – Milestones so far

Automated download of data

Automated ingestion of data into database

Automated parsing and filtering of extreme values

Connection of GRASS GIS modules to database

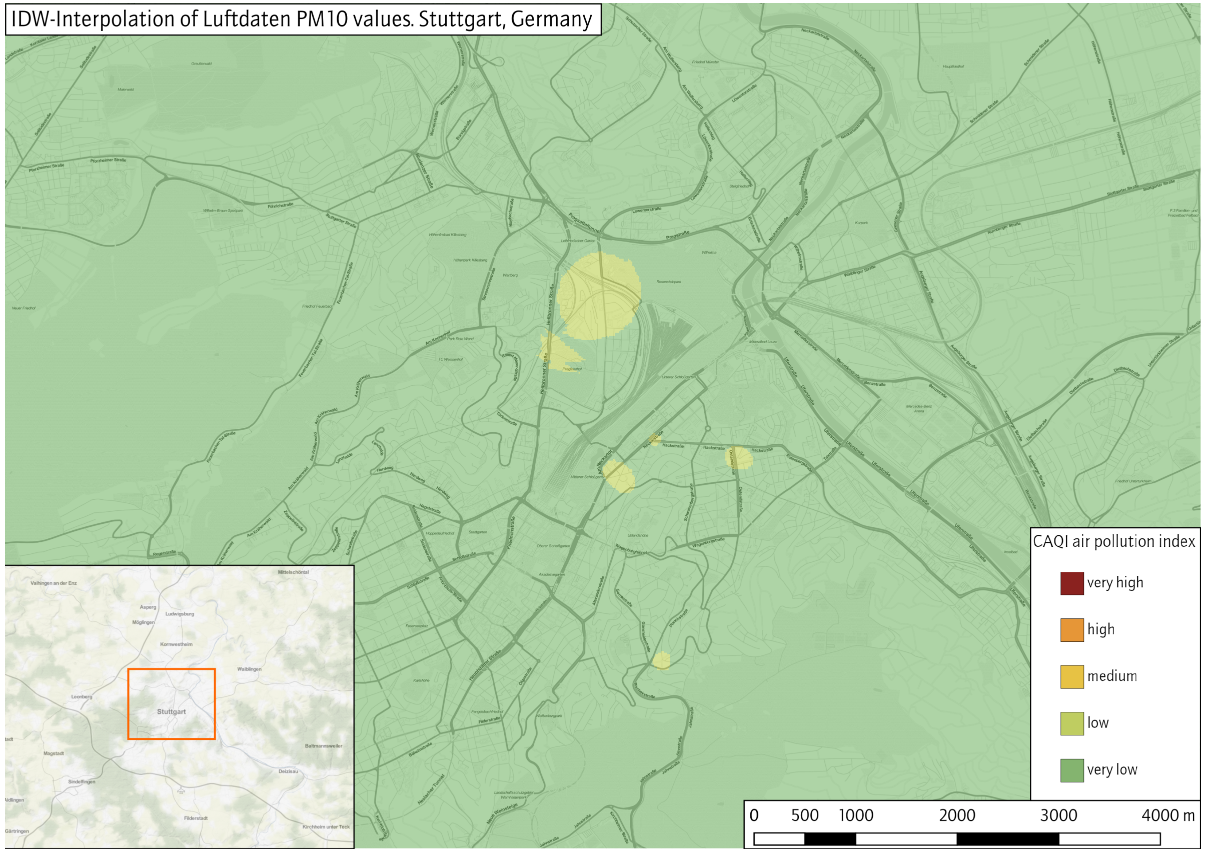
Import of stations and data

Automated interpolation of air quality data to categorized raster

Automated conversion of air quality raster to generalized GeoJson

Routing WebApp to avoid highly polluted areas

Results



Method: IDW-Interpolation

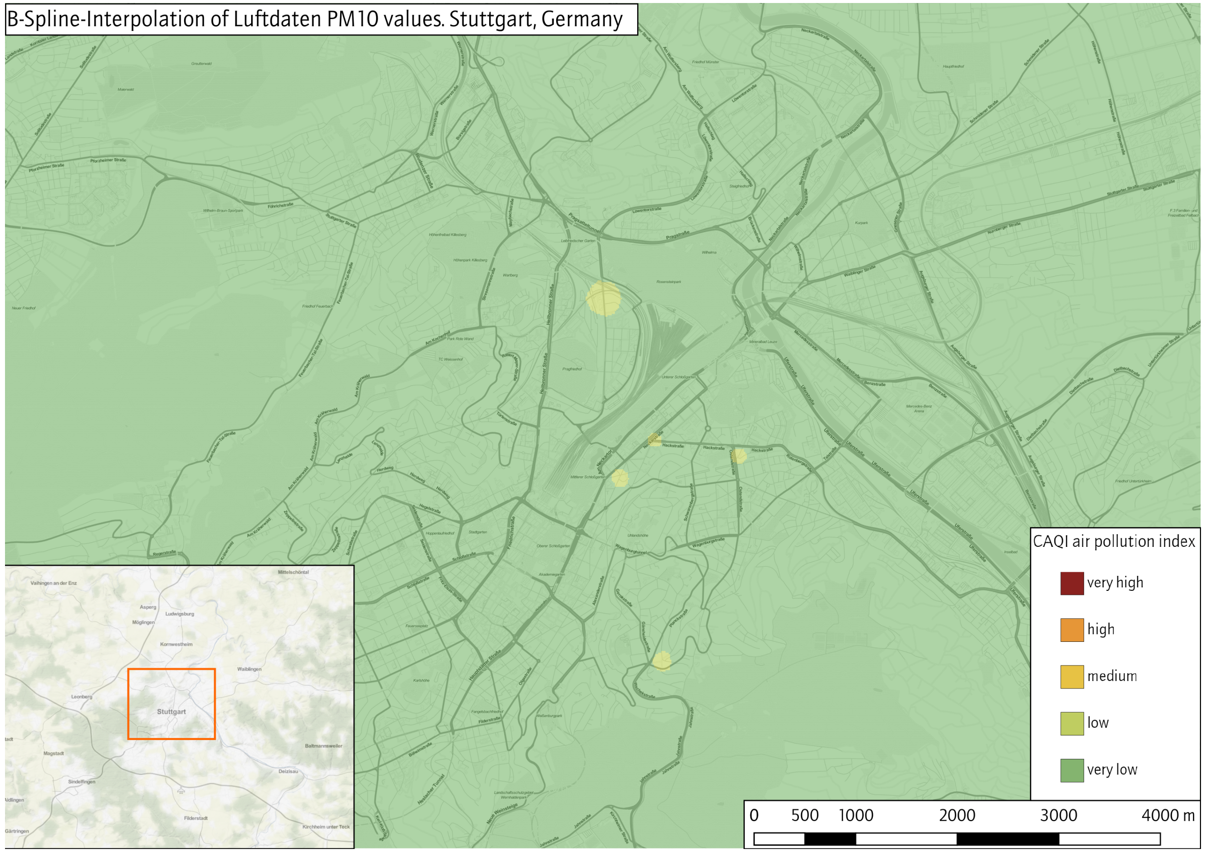
Software: GRASS-GIS

Module: v.surf.idw

Relevant Parameters:

input: luftdaten_pm10_latest
power (factor): 2.0
npoints=12

Results



Method: B-Spline-Interpolation

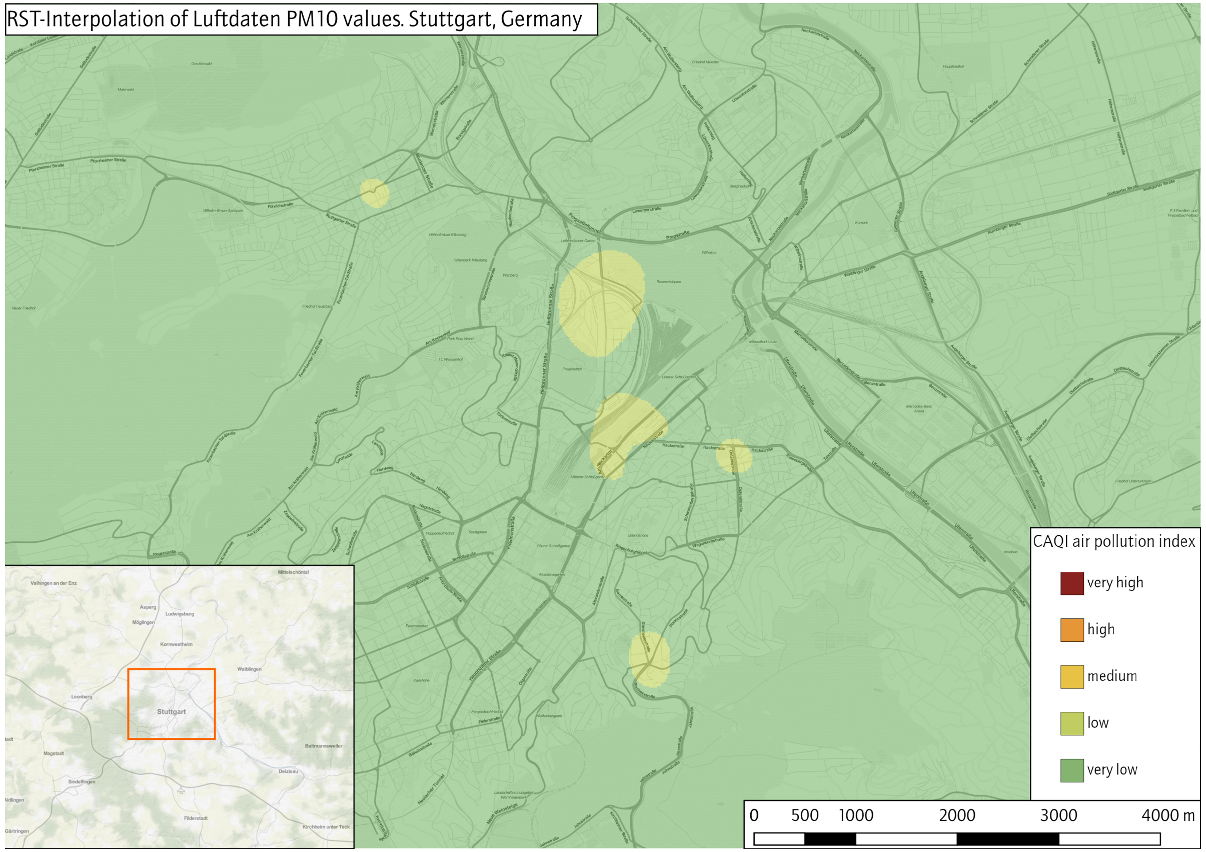
Software: GRASS-GIS

Module: v.surf.bspline

Relevant Parameters:

input: luftdaten_pm10_latest
lambda_i=0.01
ns_step=50
ew_step=50

Results



Method: RST-Interpolation

Software: GRASS-GIS

Module: v.surf.idw

Relevant Parameters:

input: luftdaten_pm10_latest
tension=40
smooth=0.5
segmax=10
npmin=50

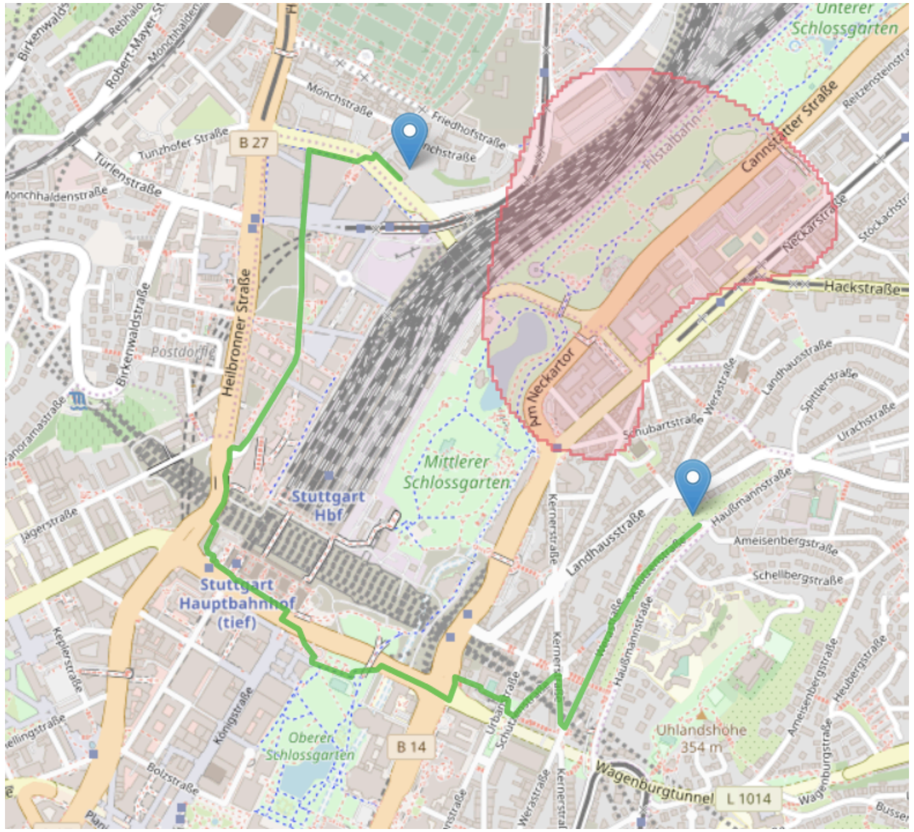
Results

VueJS Webapp using:

- vue2leaflet

- openrouteservice-js

- category geojson as avoid polygons



Results

Modelling air quality / pollution in Stuttgart

Official sensors

- Sparse network of official sensors in research area
- Modelling air pollution impossible without additional information

Luftdaten.info sensors

- Relatively large, non-systematic discrepancy between Luftdaten and LUBW sensors (cf. LUBW 2017)

BUT

- Luftdaten network *can* sense general particulate matter *trend* in Stuttgart

SO

- Citizen Science / Volunteered Geographic Information enriches model

Limitations of the analysis

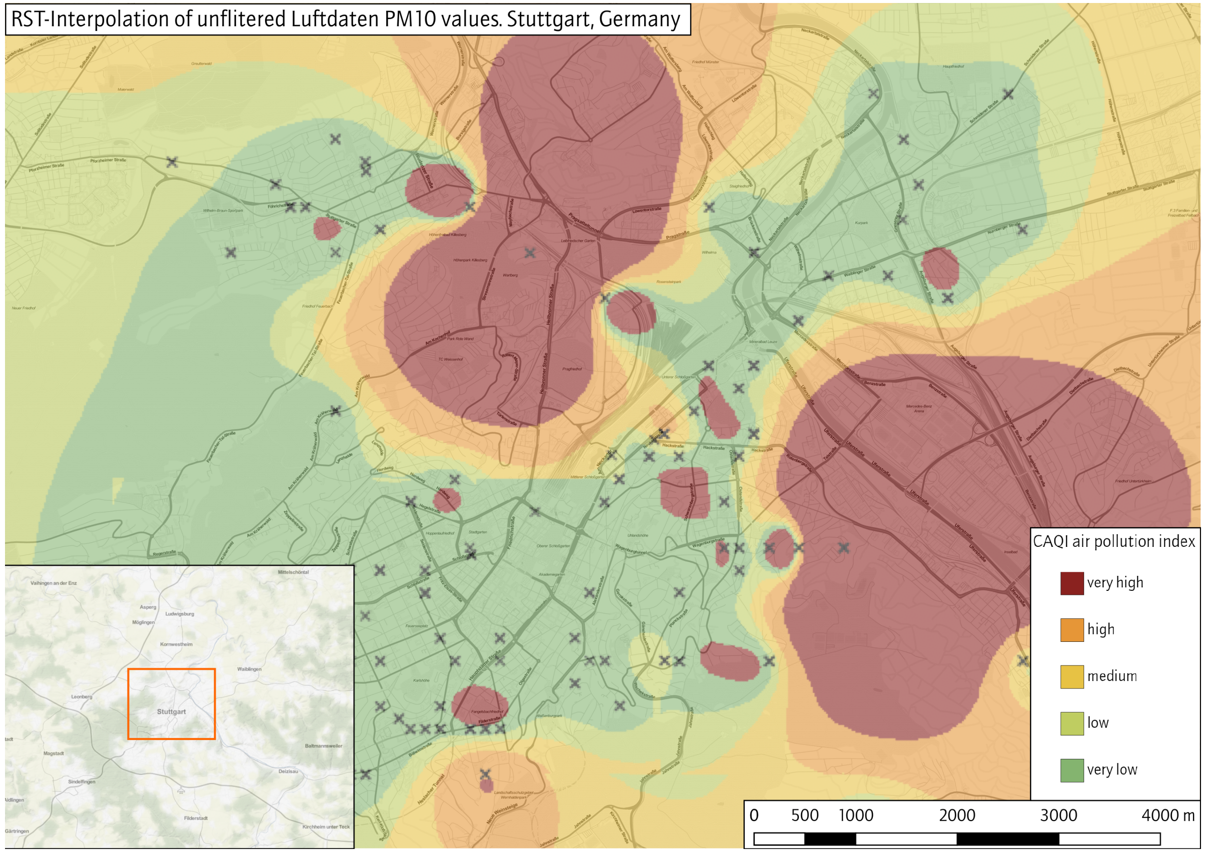
Accuracy of Luftdaten sensors
- Need to filter extreme values (both absolute and relative)
- Constant discrepancy to official measurements

Interpolation parameters are „best fit“
- No ‚correct‘ solution to model pollution rasters from sensors

Small scale of study area
- Large effects of additional factors, i.e atmospheric conditions (wind / precipitation) / land cover (buildings / parks)

Short amount of sensing time so far

Limitations of the analysis



Method: RST-Interpolation

Software: GRASS-GIS

Module: v.surf.rst

Relevant Parameters:

input: luftdaten_pm10_latest
power (factor): 2.0
npoints=12

Recommended literature

BRIGGS, D. ET AL. (1997) Mapping urban air pollution using GIS: a regression-based approach, International Journal of Geographical Information Science, 11 (7), 699-718.

LI, L. ZHOU, W. TONG, W. (2019): Spatiotemporal Analysis of Air Pollution and Its Application in Public Health. Amsterdam, Oxford, Cambridge: Elsevier. 328 p.

LUBW (2017): Messungen mit dem Feinstaubsensor SDS01. Ein Vergleich mit einem eignungsgeprüften Feinstaubanalysator. Online source: https://pudi.lubw.de/detailseite/-/publication/90536-Ein_Vergleich_mit_einem_eignungsgepr%C3%BCften_Feinstaubanalysator.pdf

MATĚJÍČEK ET AL. (2006): A GIS-based approach to spatio-temporal analysis of environmental pollution in urban areas: A case study of Prague's environment extended by LIDAR data. In: Ecological Modelling. 199 (3), 261-277.