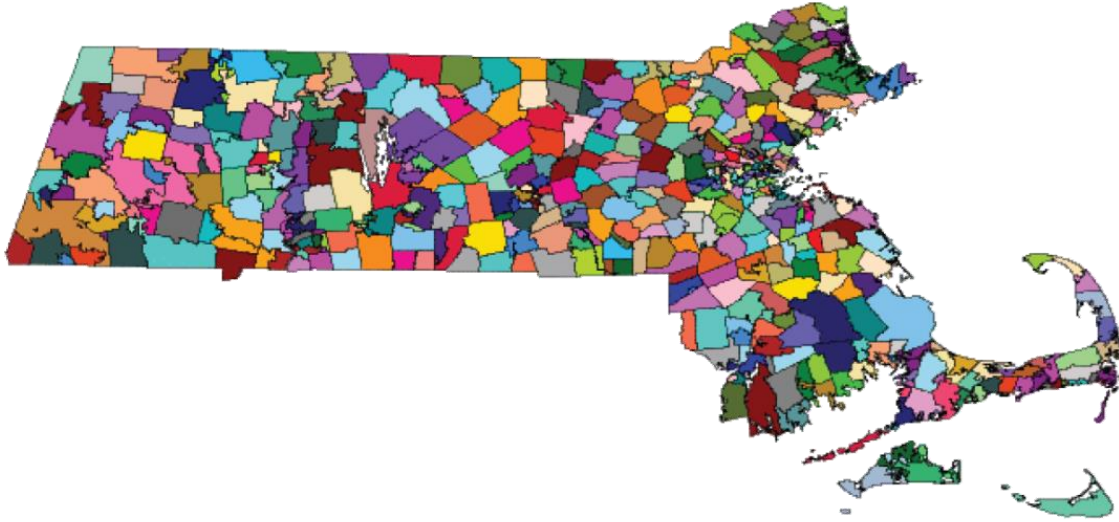Chapter 4

Geographic data

Prof. Dr. K. Verbarg

What is spatial and geographic data?

We are interested in 2D (geographic data): points or shapes (lines, polygons)

Many applications:

- o TSG Hoffenheim is using SAP HANA to provide dynamic visual analysis of player and ball movement via sensor chips embedded in the ball and player shin guards. Use of spatial processing to optimize dynamic premium ads placement based on on-the-field ball location.

- o Logistics: Display flow of goods, ships, planes,…
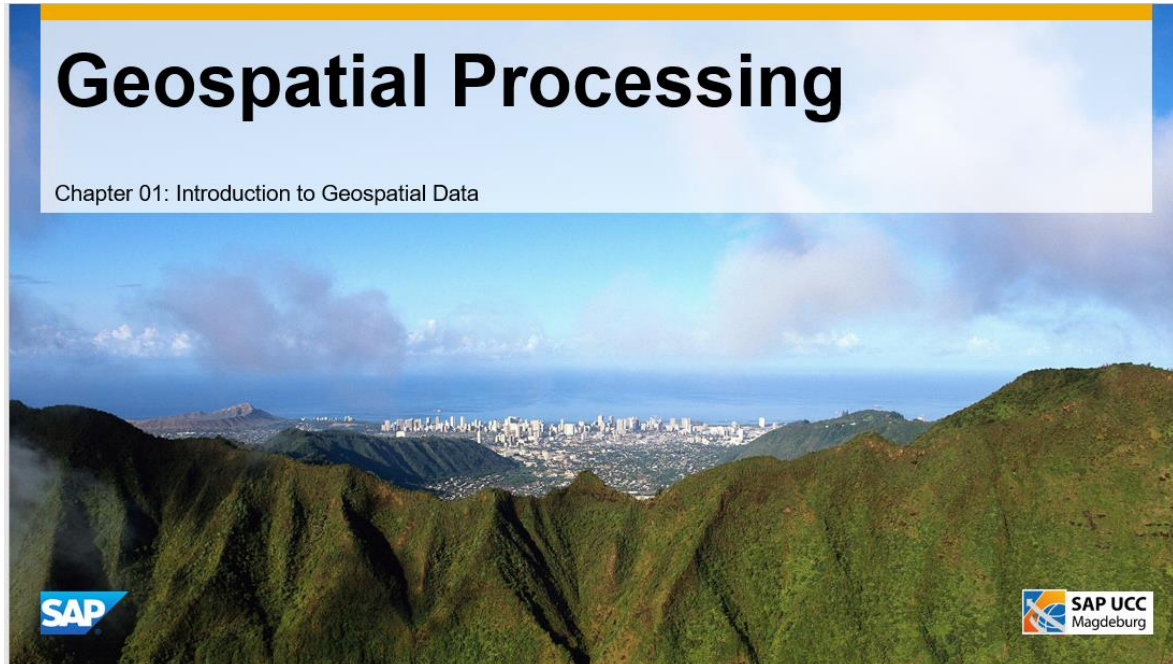
We want to do spatial operations:

- o Compute geo distances (along a great circle on the earth, or, along a road map!) or other spatial functions

- o Retrieve data within area, or nearest neighbour

- o **Geocoding**: map real addresses to geo coordinates and vice versa

- o Display such data on a map (by coordinate or using its name e.g. by country)

We refer to these slides explaining the basic geospatial terms:

Geospatial Processing

Chapter 01: Introduction to Geospatial Data

02_Geospatial_Chapter_01_Introduction_to_Geospatial_Data_Slides_en.pptx

Bottom line:

o Geo position has latitude and longitude

o Coordinate system is WGS84 with SRID 4326

o For SQL, there is a standard ISO/ IEC 13249-3 describing
  – data types (ST_POINT,…)
  – formats (latitude/longitude, WKT well-known text, ESRI shapefile, GeoJSON, SVG)
  – SQL commands to do spatial operations

Our mission today...

Geographic data is precious and expensive.

- o **google**
- o **here-maps** (Audi, BMW, Daimler, Bosch, Intel...) – formerly Navteq, Nokia
- o https://www.openstreetmap.org
- o Microsoft bing

You can do geo processing in Python using openstreetmap data. Overview: https://pythongis.org
Or: geopandas project.

Usually, you will have to use <u>cloud</u> services for working with geo data.

As an example, we do geocoding of addresses with the help of a google API.

→ *shows connecting to a cloud service*

→ *shows geocoding*

Create your student account

Here is the URL you will need to access in order to request a Google Cloud coupon. You will be asked to provide your school email address and name. An email will be sent to you to confirm these details before a coupon is sent to you.

Student Coupon Retrieval Link

• You will be asked for a name and email address, which needs to match your school domain. A confirmation email will be sent to you with a coupon code.

• You can request a coupon from the URL and redeem it until: 2/3/2023

• Coupon valid through: 10/3/2023

• You can only request ONE code per unique email address.

Students hereby receive 50 USD credit.

Activate geocoding API

See https://developers.google.com/maps/documentation/geocoding/overview

Follow the steps "Einrichtung in der Cloud Console":

o Create a project (make sure to have the correct "Abrechnung" account connected (not the private credit card)

o Activate geocoding API and get a key

o Check you have 50 USD credit (see "Abrechnung")

Menu "Geocoding-Anfrage und –Antwort" gives a description of using the service.

Menu "Nutzung und Abrechnung" shows the theoretical costs.

Eventually do some geocoding

Use API in python

See https://github.com/googlemaps/google-maps-services-python

- Check that `googlemaps` is installed using the "Anaconda Prompt":
  `conda install -c conda-forge googlemaps`
- Test the API

```
import googlemaps
%xmode minimal

Exception reporting mode: Minimal

API_KEY = 'AIzaSyDPFE06t5QqWpAX3       79dnRP7DVkw'
gmaps = googlemaps.Client(key=API_KEY)

result = gmaps.geocode(address='NORREGADE 10 1165 KOBENHAVN', region='dk')

ApiError: REQUEST_DENIED (You must enable Billing on the Google Cloud Project at https://
t/_/billing/enable Learn more at https://developers.google.com/maps/gmp-get-started)
```
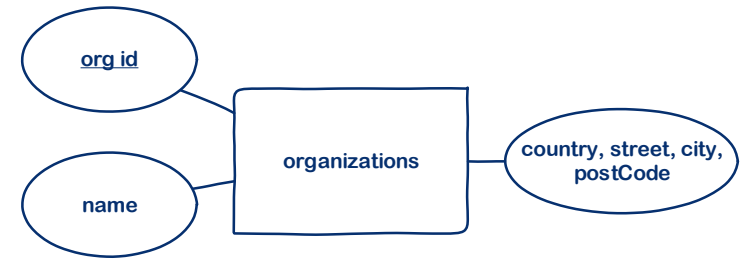
Implement

- Download CORDIS organisation H2020 data as CSV (European patent data)

- Extract relevant columns (we don't need projects)
- Remove duplicates?
- Prepare region and address
- Call geocode (test only with a single address to avoid spoiling all your credits at once!)
- Do the entire dataset and save the results to CSV

Optional tasks

o Are there errors, e.g. addresses not converted successfully?

o Are there addresses which are not unique (multiple results of geocoding)?

o What is the distance between our geocoding result and the "geolocation" contained in the original dataset?

o display data on a map (SAP Analytics Cloud or google maps)