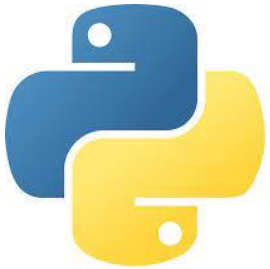


Python
Introduction
Course

Python Course
GIS-introduction

Ronan Paugam, Eulàlia Planas

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

GIS - Geographic Information System

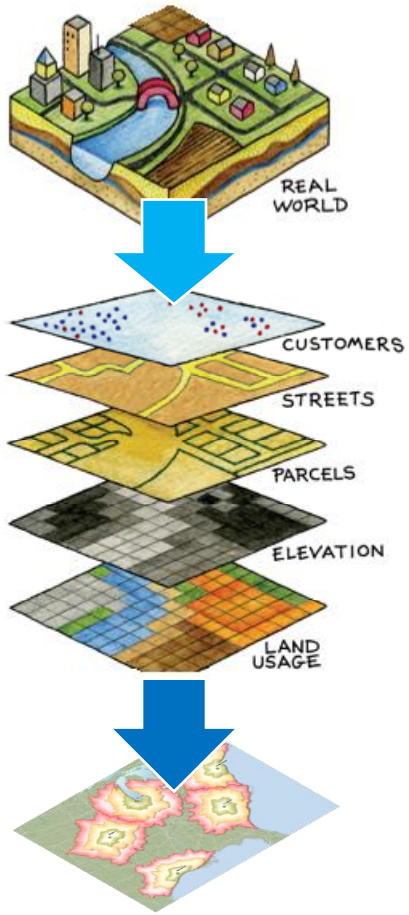
Geographic Information System - GIS

"A system for capturing, storing, checking, integrating, manipulating, analyzing and displaying spatial data"

GIS recreates real world spatial data

as digitized themed data "layers" (e.g. locations, boundaries, infrastructure, socioeconomic hydrology, land use/cover)

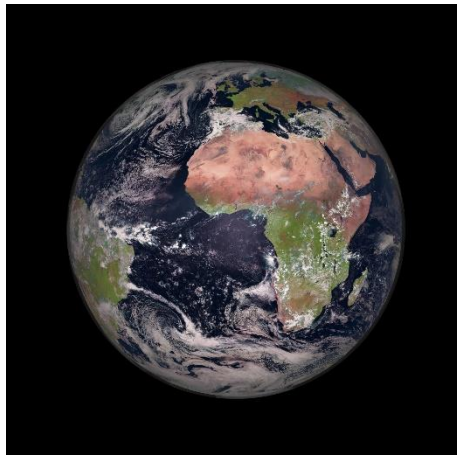
assembled in any combination and overlaid for analysis



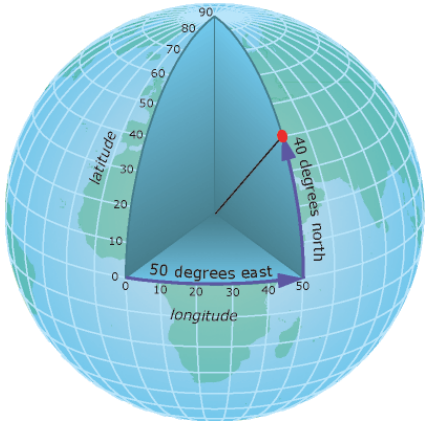
GIS: Coordinate System

Geographic Coordinate System (GCS)

- angular unit of measure
- prime meridian
- datum based on a spheroid



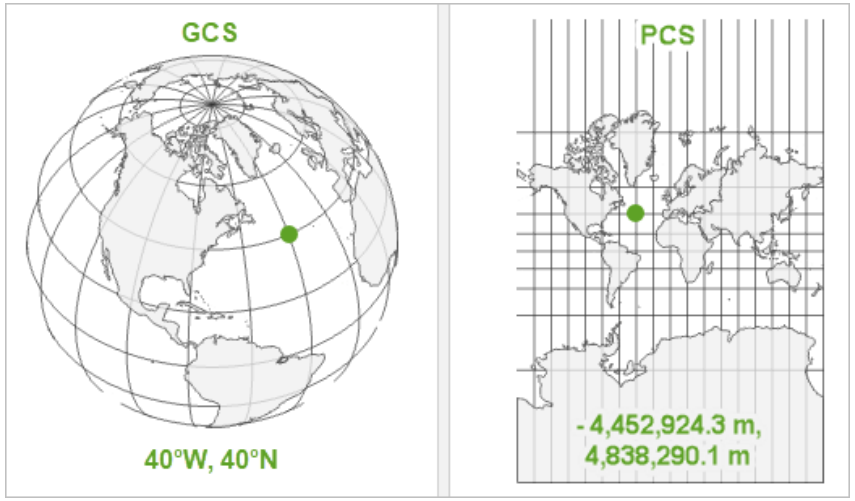
GCS



WGS84 is the most widespread

Projected Coordinate Systems (PCS)

from the sphere to the map

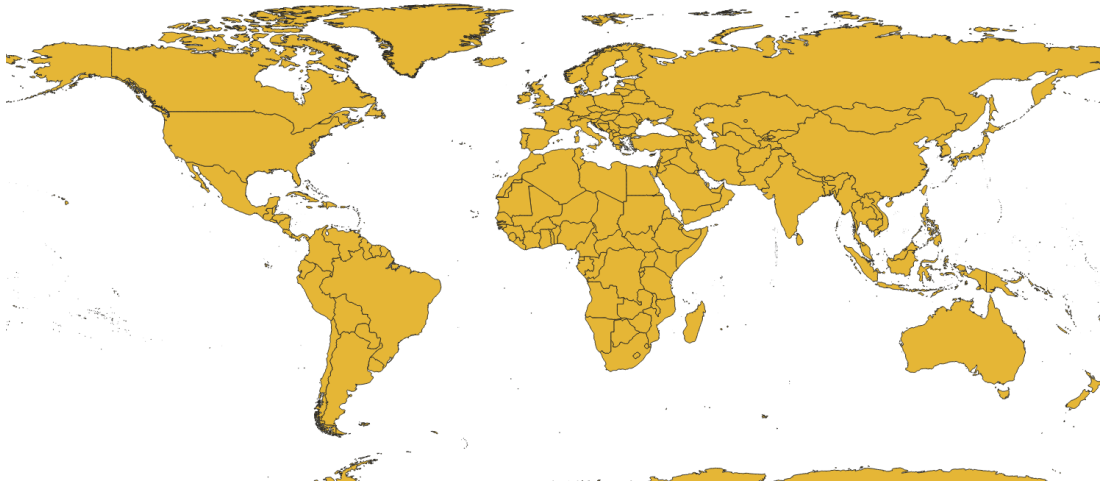


PCS creates local distortion and affect area, angle and distance

- The GCS tells your data where to draw. The PCS tells the map how to **stretch the GCS out flat.**
- Which **GCS you choose** depends on where you are mapping.
- **Which PCS you use** depends on where you are mapping, but also the nature of your map, for example, should you distort area to preserve angles, or vice versa?

GIS: Coordinate Reference System CRS=GCS+PCS

EPSG:
public registry of geodetic datums, spatial reference systems, Earth ellipsoids, coordinate transformations and related units of measurement, originated by a member of the European Petroleum Survey Group (EPSG) in 1985



EPSG 4326: grid longitude = -180 to 180 and latitude = -90 to 90
- Projection: Plate Carree
- GCS = WGS84
- used by the GPS system

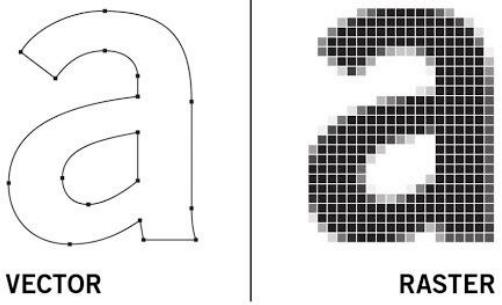
EPSG 3035: conserve distance in Europe (<1m)
- Projection: Lambert Azimuthal Equal Area
- GCS = WGS84

GIS: Data Representation

Information can be stored in **vector** or **raster** format.

Vectors are composed of coordinates

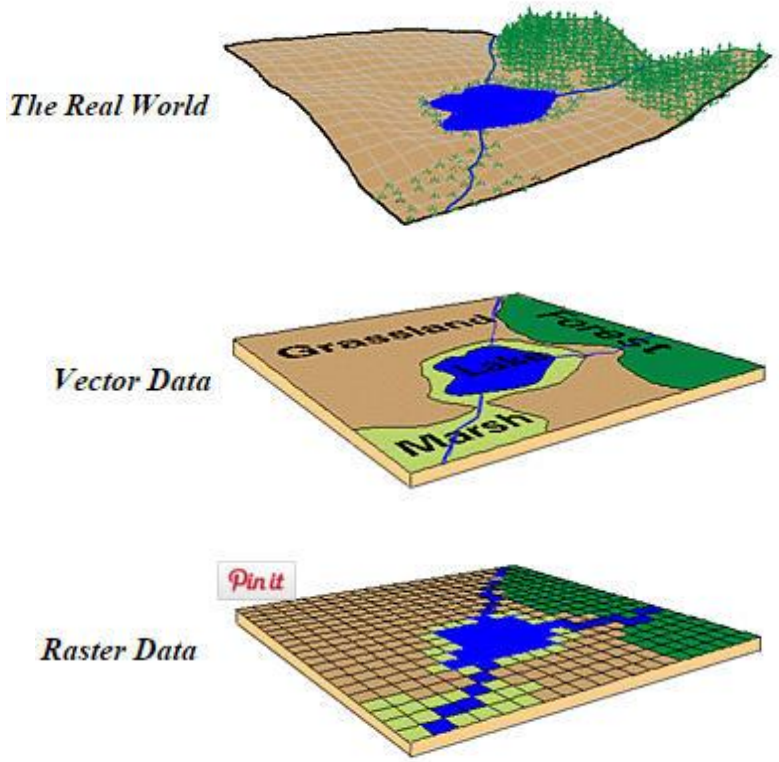
Raster's are composed of pixels



These are often used for variables with:

defined borders, e.g. manmade

continuous surface, e.g. environmental



Geospatial or coordinate data can be represented in two different data formats:

Vector:

e.g. points, lines, and polygons

Fire Font can be stored as polygon

Raster:

e.g. row and column matrix

Fuel Class Map can be stored as raster

GIS: Software

Type	Analysis Power	Example(s)
Geobrowser	**Weak** (mainly only to display data)	Google Maps, Google Earth, Apple Maps, Waze, etc.
Web-based	**Medium** (able to upload additional data, customize display, and perform basic analyses)	Carto, ArcGIS Online, Mapbox, Google MyMaps, etc.
Desktop	**Strong** (installed locally, provides full control of map creation, and perform advanced analyses)	ArcGIS Pro **QGIS** **Pyhton librairy**

QGIS:

- Free, open-source tool
- Runs on any operating system
- Smaller program that will not affect performance of your computer
- Many available tools, but lacking some for specific functions, such as network analysis (i.e. routing) and spatial statistics
- Basic tutorials by QGIS developers and users
- Tools can be developed by anyone so performance and documentation is inconsistent.
- Support via forums

Practical

Here you have a series of example showing basic example of GIS task in pyhton

Python and GIS

A Geographic Information System (GIS) is a computer system that analyzes and displays geographically referenced information. It uses data that is attached to a unique location.

Data are generally of two form:

- **raster data**: it is made up of pixels (also referred to as grid cells). They are usually regularly spaced. Rasters often look pixelated because each pixel has its own value.
- **vector data**: it is made of vertices and paths and therefore not attached to grid of pixels.

In this course, several notebooks shows example of how to deal with raster and vector data. See the list below:

- 05a-GIS-rasterio.ipynb shows an example of loading and plotting a raster image using the `rasterio` and `matplotlib` libraries.
- 05b-GIS-geopandas.ipynb shows an example of how to create vector data and plot them using `geopandas` and `matplotlib`
- 05c-GIS-vector2raster.ipynb shows an example of projecting a vector polygon onto a raster using `rasterio` and `geopandas`.
- 05d-GIS-netcdf-cartopy.ipynb shows an example of loading 2D map stored in netcdf file using `netcdf4` and plotting them on projected map using `cartopy`.