INTRODUCTION AND TOOLS

Master in Environmental Management of Mountains Areas

ADVANCED GEOMATICS

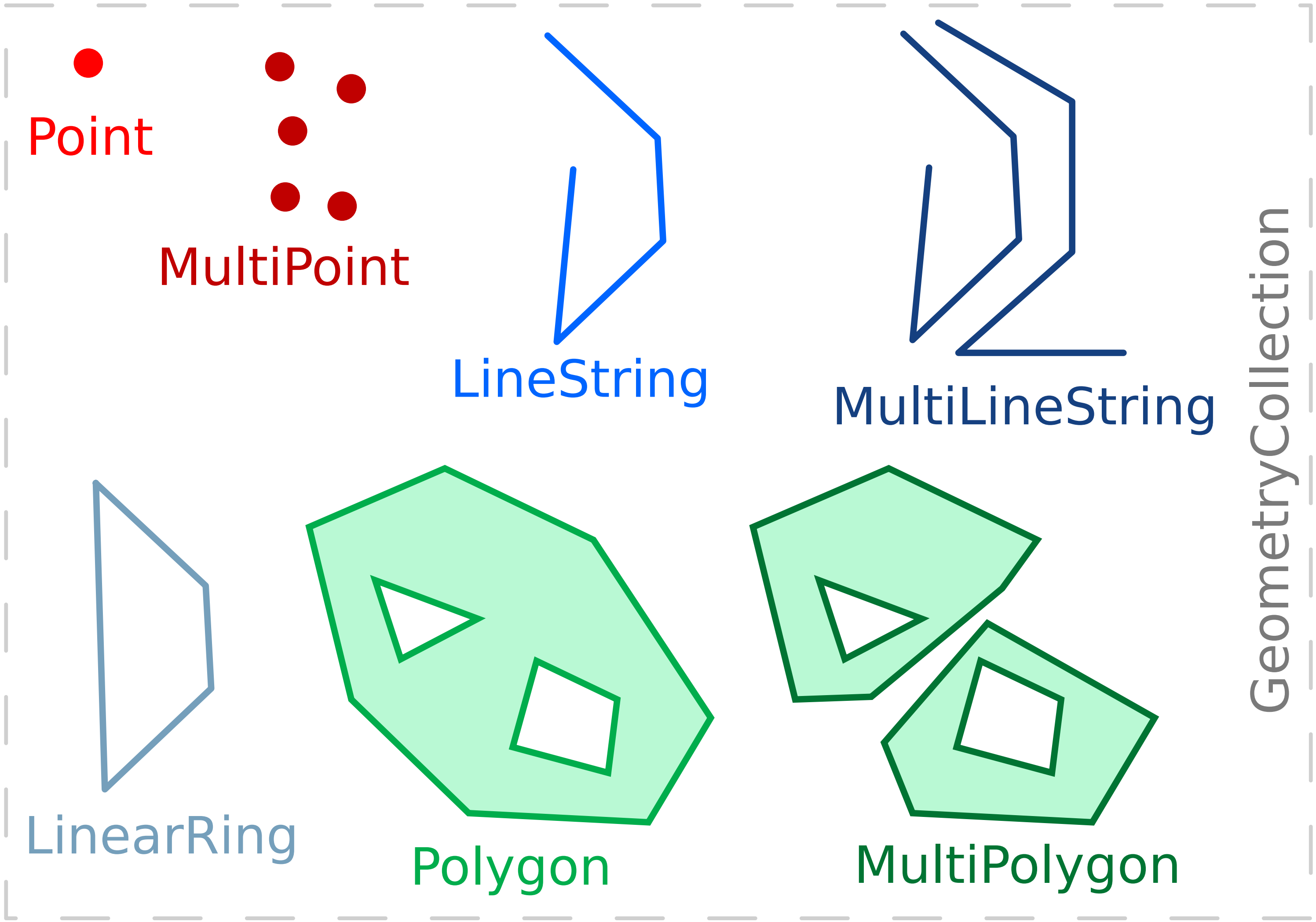
Andrea Antonello - Free University of Bolzano

March - June 2024

LET'S FIRST FIND A COMMON LANGUAGE

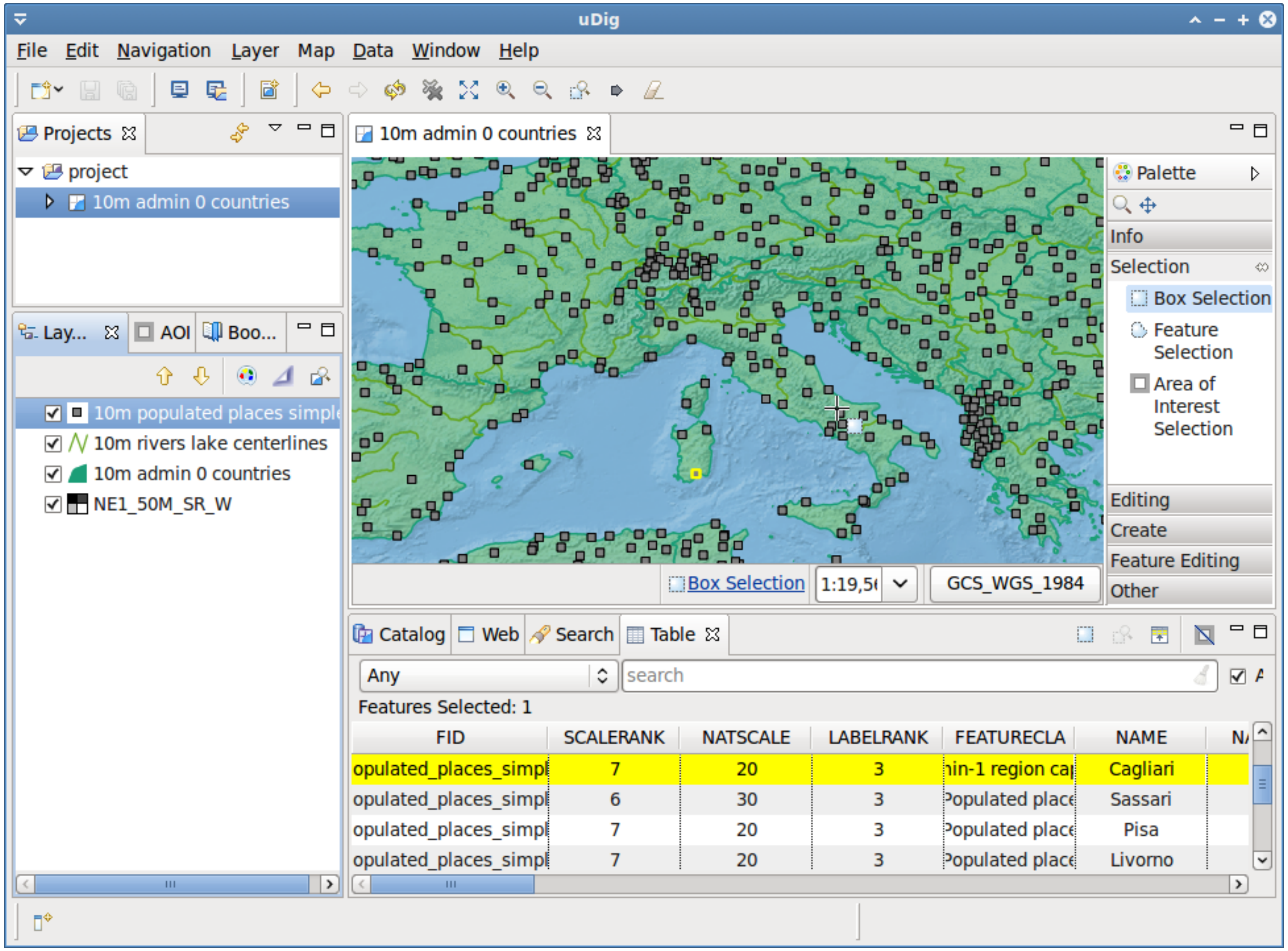
THE MAIN VECTOR GEO-OBJECTS

The main spatial data types that are usually dealth with are:



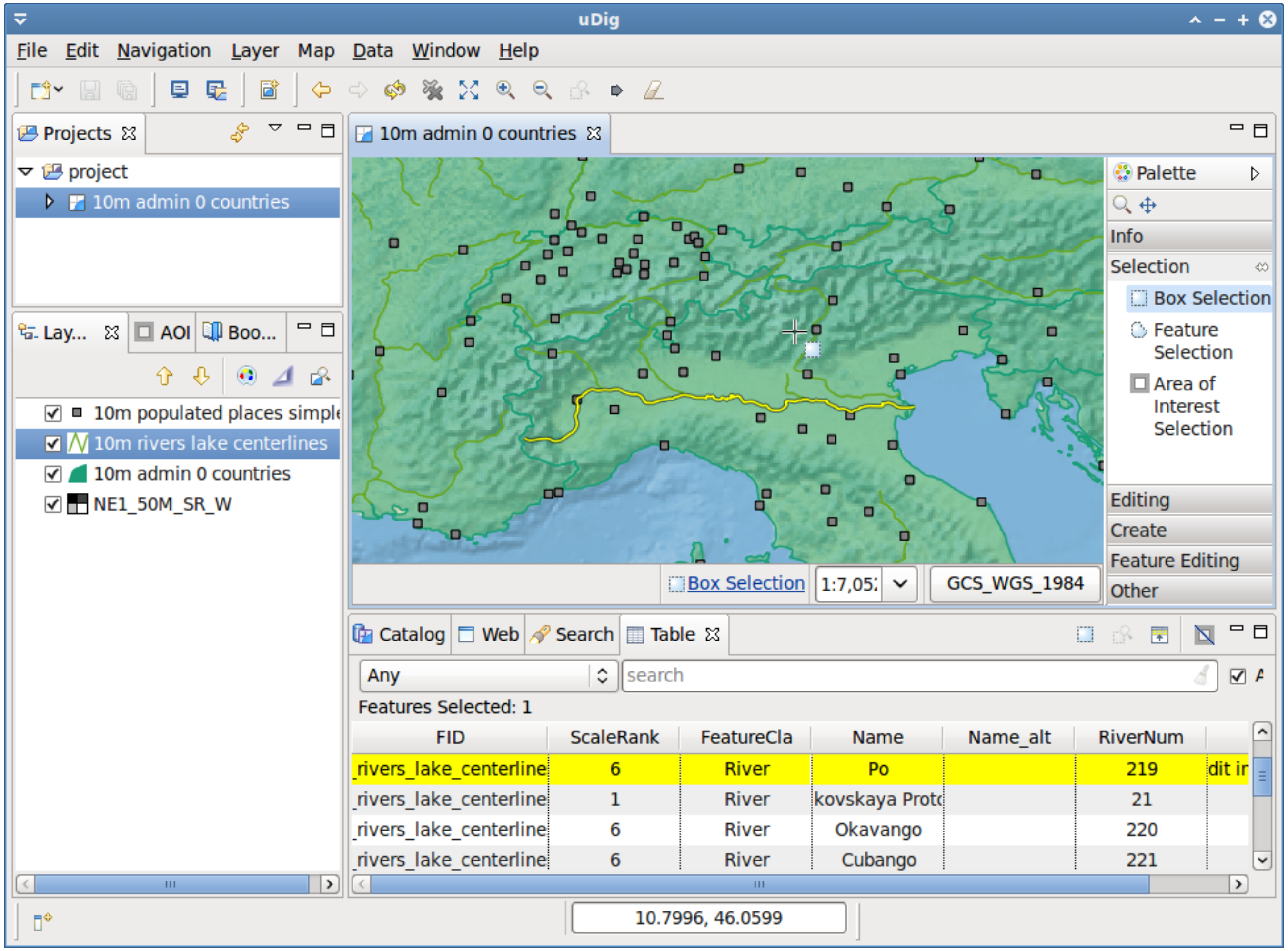
(MULTI)POINT

A Point models a single Coordinate, a MultiPoint models a collection of points.



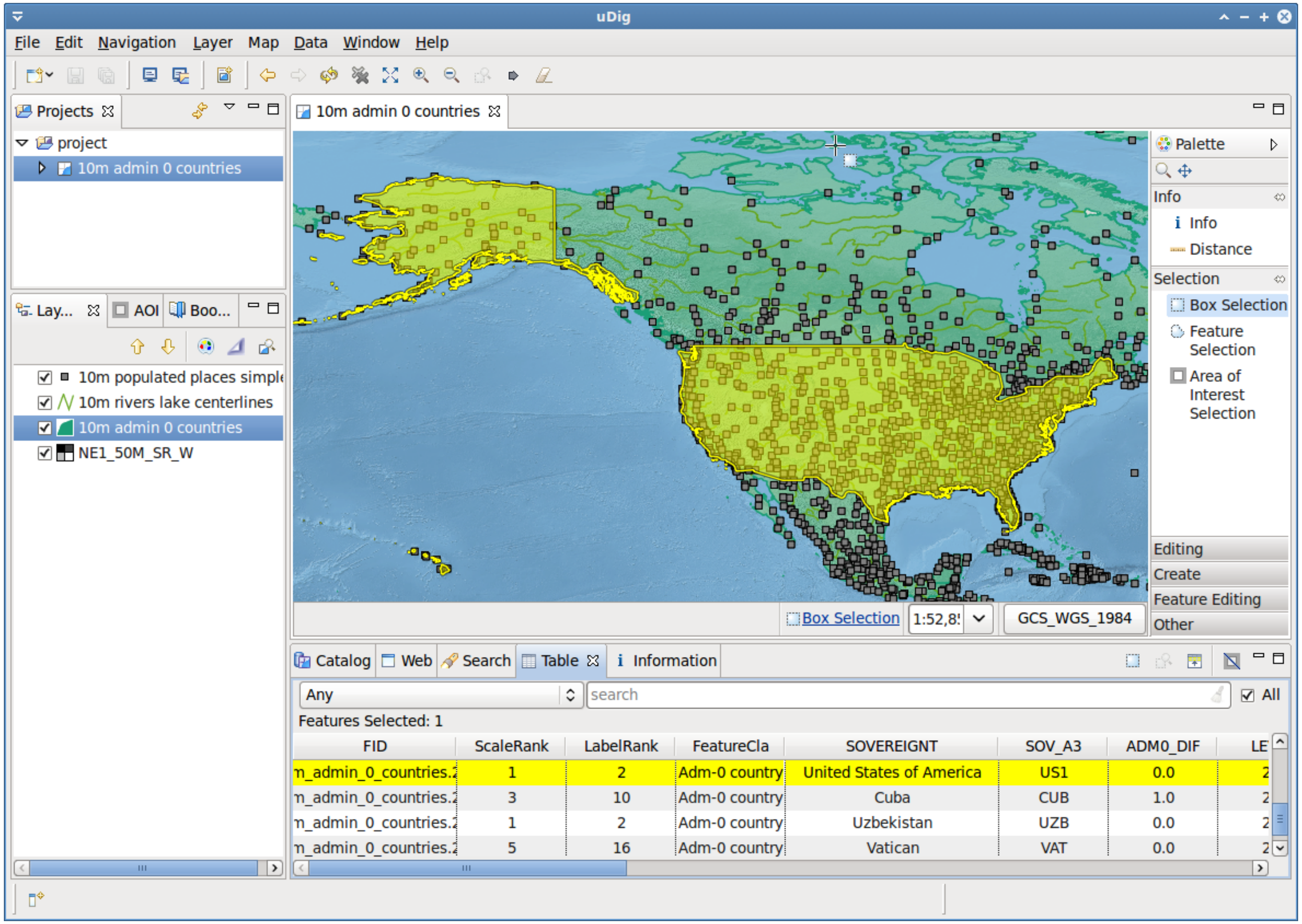
(MULTI)LINESTRING

The LineString is what we generally call line. It has a length, but 0 area.



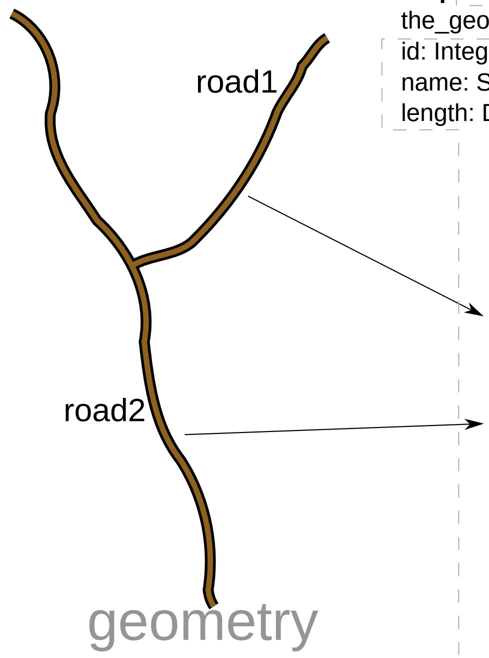
(MULTI)POLYGON

Polygons have a length (perimeter) and an area. They can also have holes.



VECTOR DATA: FEATURE(TYPE)

The **Feature** represents probably the most central object for GIS applications. The **Schema** or **FeatureType** can be seen as the blueprint of the data. Vector data are composed of a geometry part and an attribute table.



SimpleFeatureType:
the_geom: LineString
id: Integer
name: String
length: Double

id	name	length
1	road1	56.4
2	road2	120.0

attributes

FILTERS

A **Filter** defines a constraint that can be checked against an object.

A filter can be seen as the WHERE clause of an SQL statement. It can apply both to the alphanumeric values of an attribute table as well as to the geometry.

One example could be: **give me all the cities of Canada that count more than 10000 inhabitants**.

COMMON QUERY LANGUAGE

The CQL (Common Query Language) is a query language created by the OGC for the Catalogue Web Services specification and is used to define expressions and filters.

A tutorial about the use of CQL can be found here.

Examples:

- CITY = 'Nelson'
- ATTR1 < (1 + 2 / 3) * 4
- ATTR1 < abs(ATTR2)
- ATTR1 < 10 AND ATTR2 < 2 OR ATTR3 > 10

STYLE

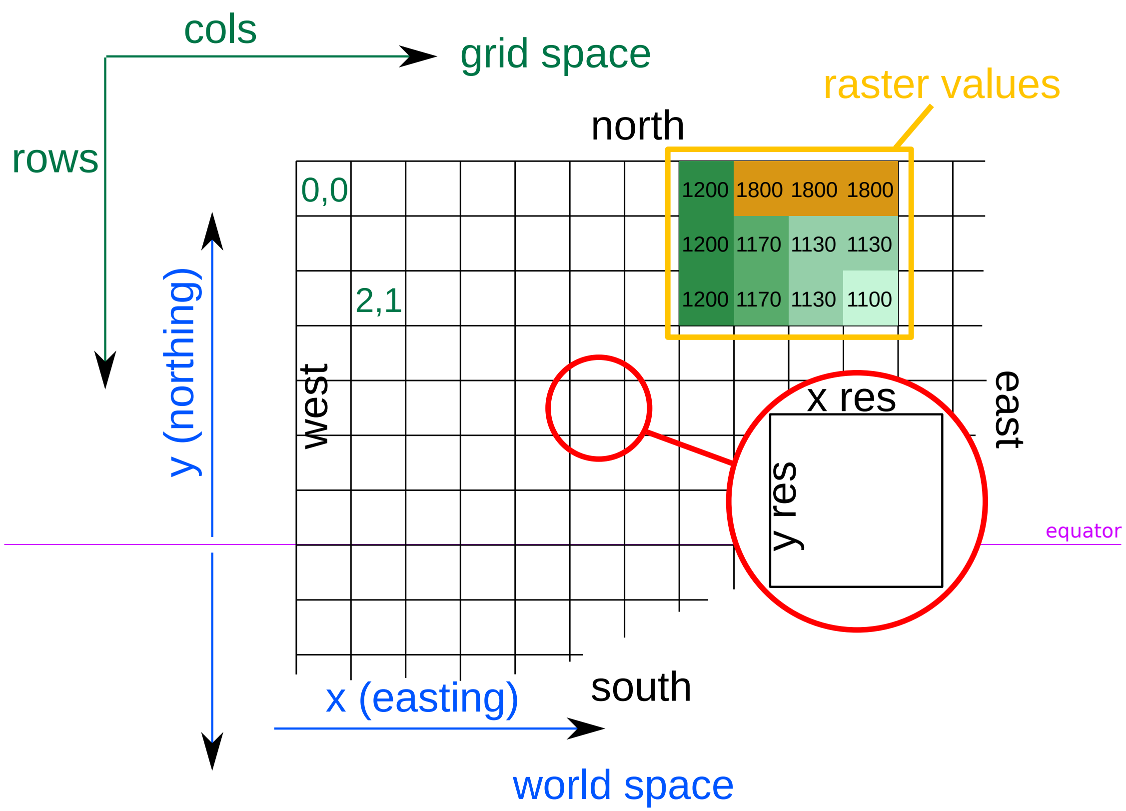
Style is that part that allows us to make maps look pretty and get the needed symbolization and coloring of the contained data. The OGC defines the **Styled Layer Descriptor** (SLD) as the standard to style maps.

gvSIG and QGIS support simple SLD. Geoserver and uDig support also complex SLD very well.

RASTER DATA

RASTER DATA: GRIDCOVERAGE

A **GridCoverage** is what in the real world we usually call **Raster** or **Grid**, i.e. a rectangular regular grid of pixels, each containing a value. The following schema contains the main definitions we will use:



COORDINATEREFERENCESYSTEM (CRS)

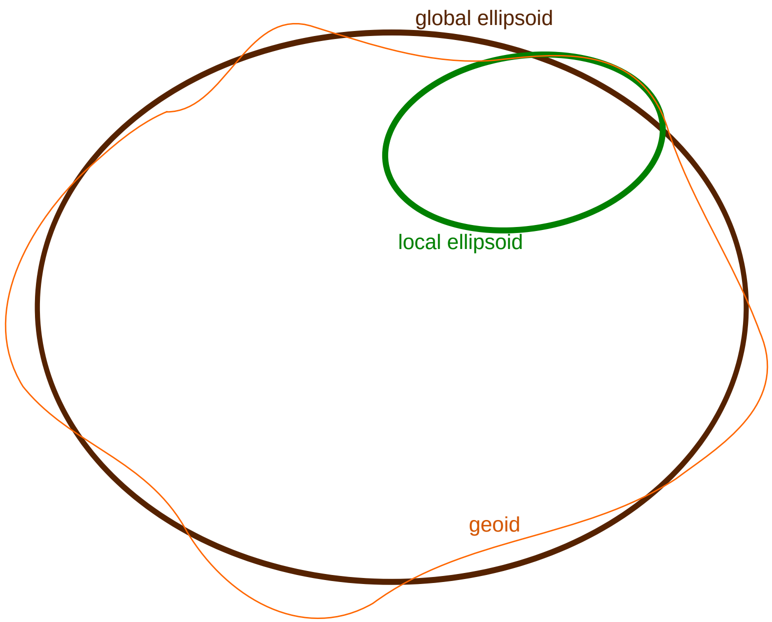
CRS

WIKIPEDIA: "A spatial reference system (SRS) or coordinate reference system (CRS) is a **coordinate-based** local, regional or global **system used to locate geographical entities**. A spatial reference system **defines a specific map projection**, as well as **transformations between** different spatial reference systems. Spatial reference systems are defined by the OGC's Simple feature access using well-known text, and support has been implemented by several standards-based geographic information systems. Spatial reference systems can be referred to using a SRID integer, including EPSG codes defined by the International Association of Oil and Gas Producers."

THE DATUM

The datum is a reference surface from which measurements are made (Wikipedia).

Datums can be **local**, which are locally orientated ellissoid (no deviation on the vertical, locally tangent), or **global**, which are used to cover the whole globe and designed to support satellitar measurements.



EXAMPLE DATUMS

- **Roma 40**

  local datum based on Hayford ellipsoid, with prime meridian on Monte Mario (EPSG:3003/4)

- **European Terrestrial Reference System 1989**

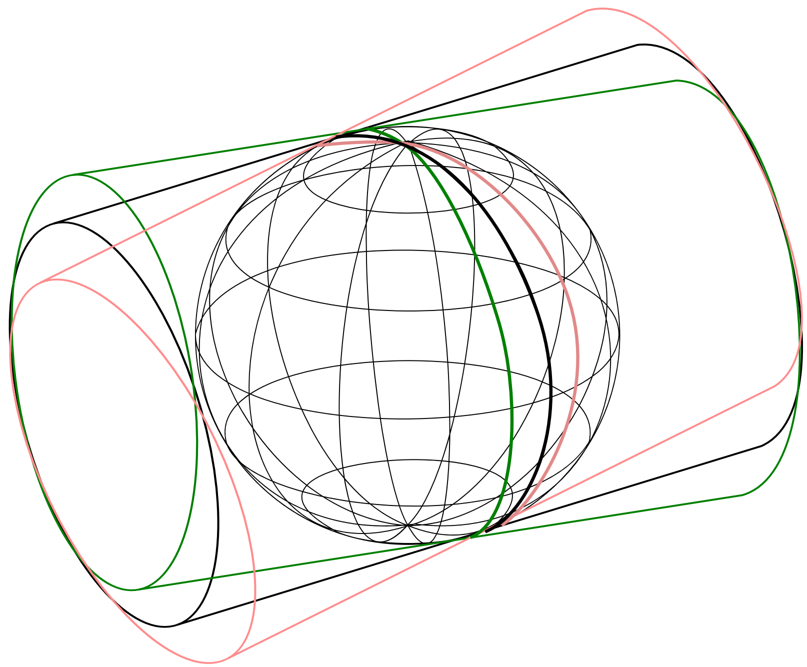
  local datum based on GRS 1980 ellipsoid, with prime meridian in Greenwich. (ex. EPSG:25832)

- **World Geodetic System WGS84**

  global datum with origin on the earth's mass center. Used for example in the classic GPS CRS (EPSG:4326) and in the WGS 84 / UTM (ex. EPSG: 32632)

UTM

UTM (The Universal Transverse Mercator) maps the Earth with a transverse cylinder projection using 60 different meridians, each of which is a standard "UTM Zone". By rotating the cylinder in 60 steps (six degrees per step, ~800Km) UTM assures that all spots on the globe will be within 3 degrees from the center of each projection.



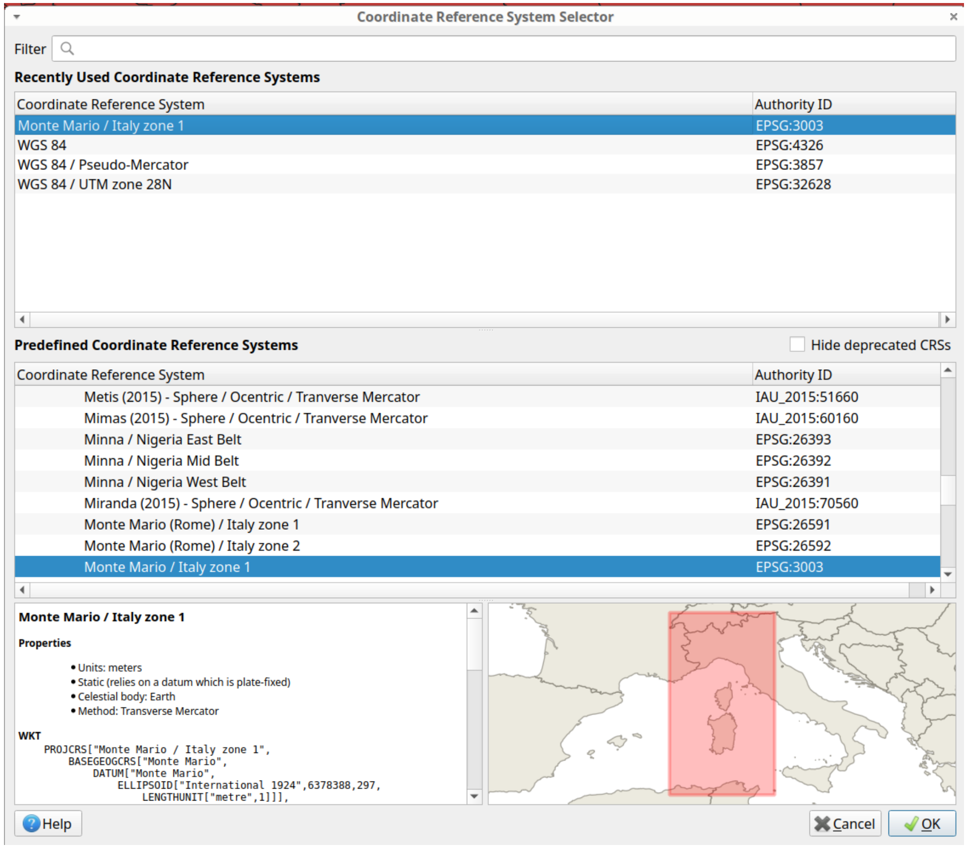
COORDINATE REPROJECTION AND TRANSFORM

Often **reproject** and **transform** are used the same way without much care. There is a big difference though.

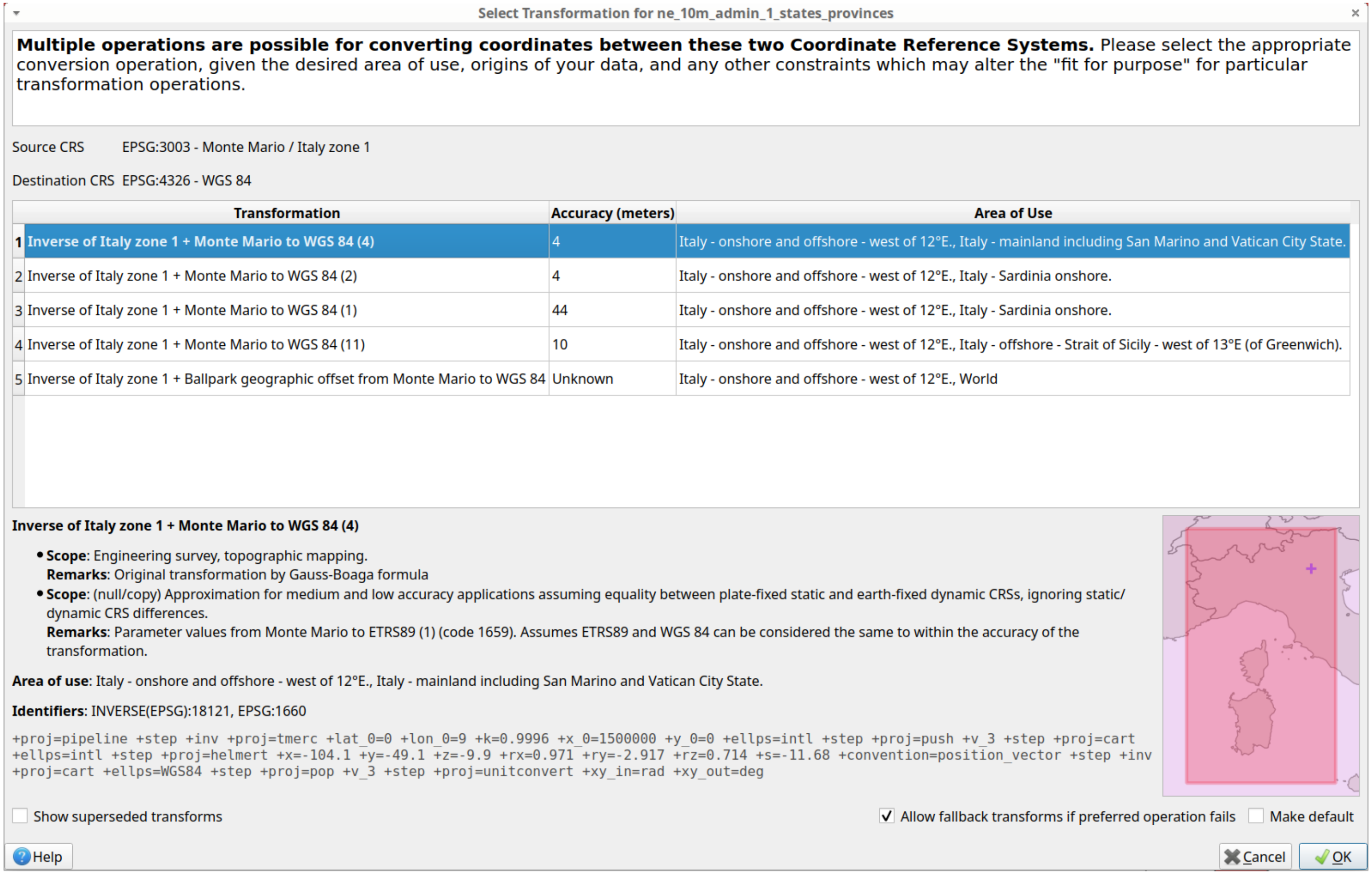
- reproject: This is what we would call coordinate transform (CT). A CT can be resolved in a well defined mathematical manner that doesn’t lead to precision loss (even if usually there is some minor due to data precision and roundings).
- transform: This is what we could call datum transform. Since datums are local approximations of the geoid, transformations between datums are based on statistical methods and lead most of the times to precision loss.

COORDINATE TRANSFORM IN QGIS

In QGIS when adding datasets to a layer it is possible to select a projection to properly render the layer through on-the-fly reprojection.

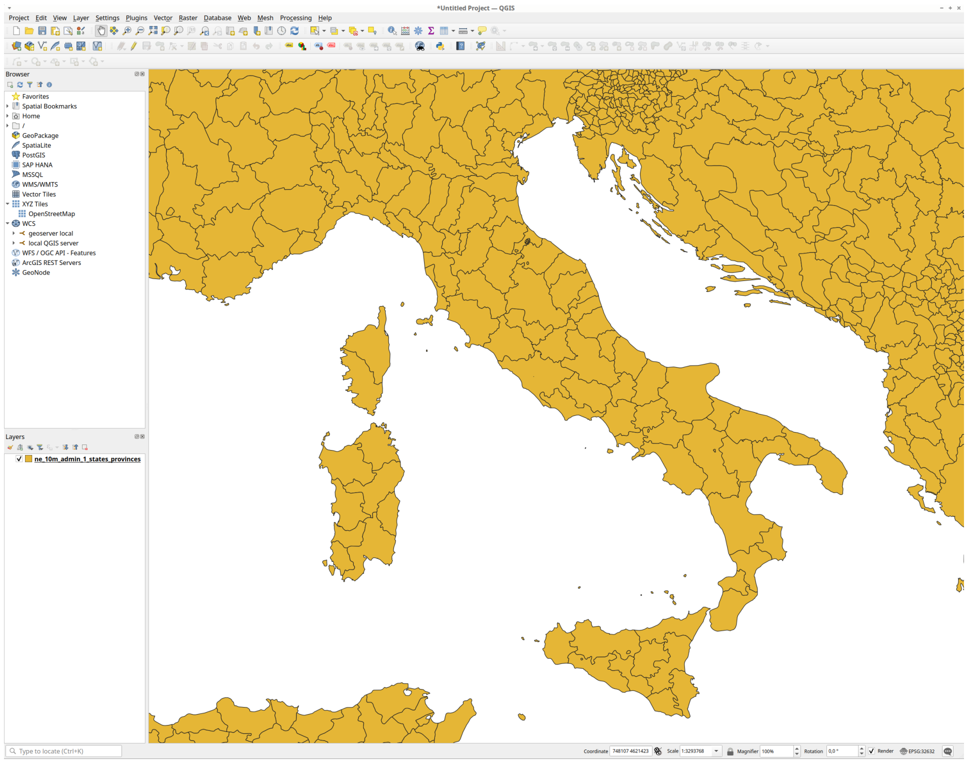


In cases in which the datum is different (ex. Monte Mario / Italy zone 1: 3003 → Lat/long WGS84: 4326), if the projection has the necessary information, the proper transformation parameters can be choosen:

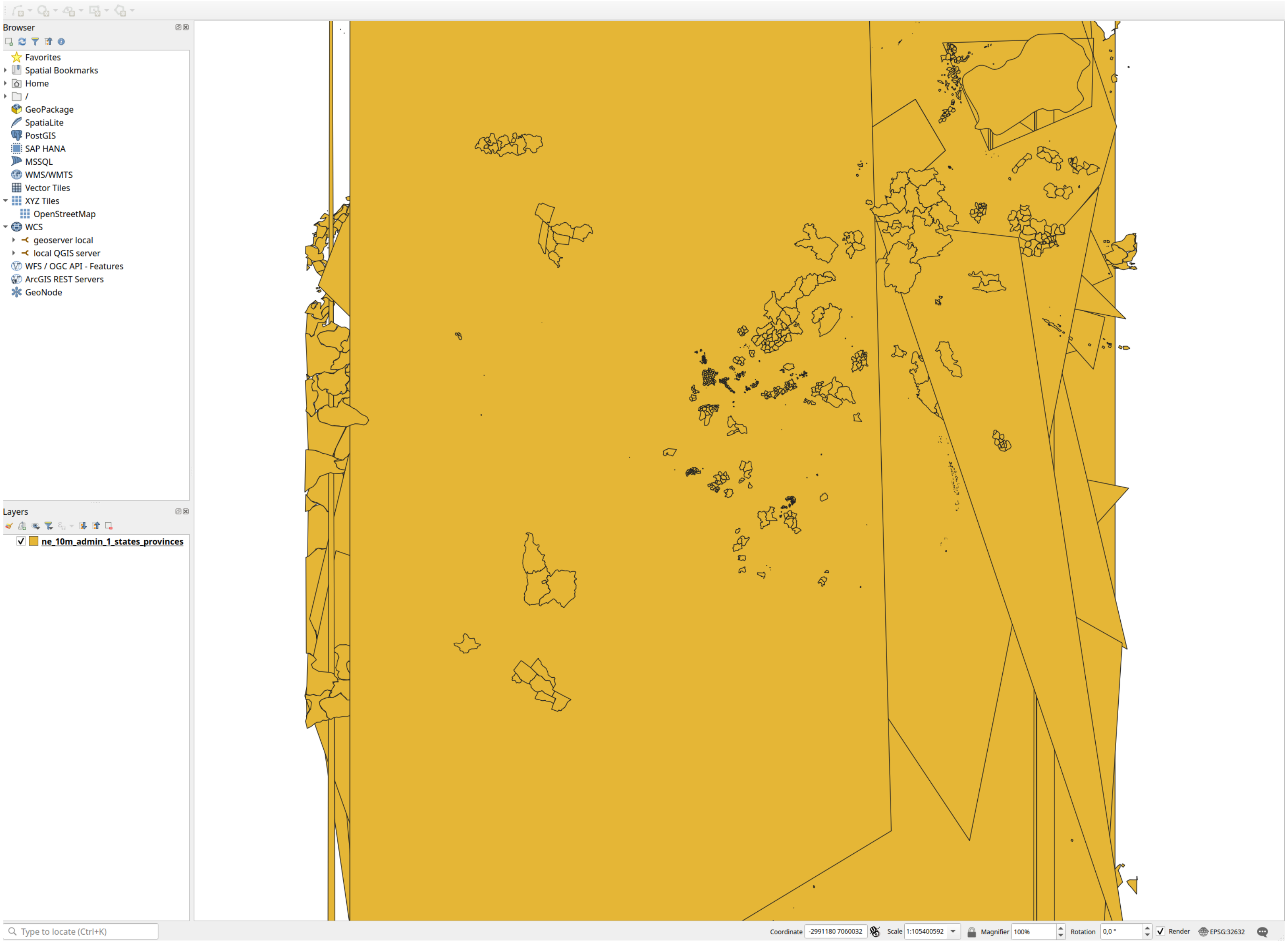


CRS MESS

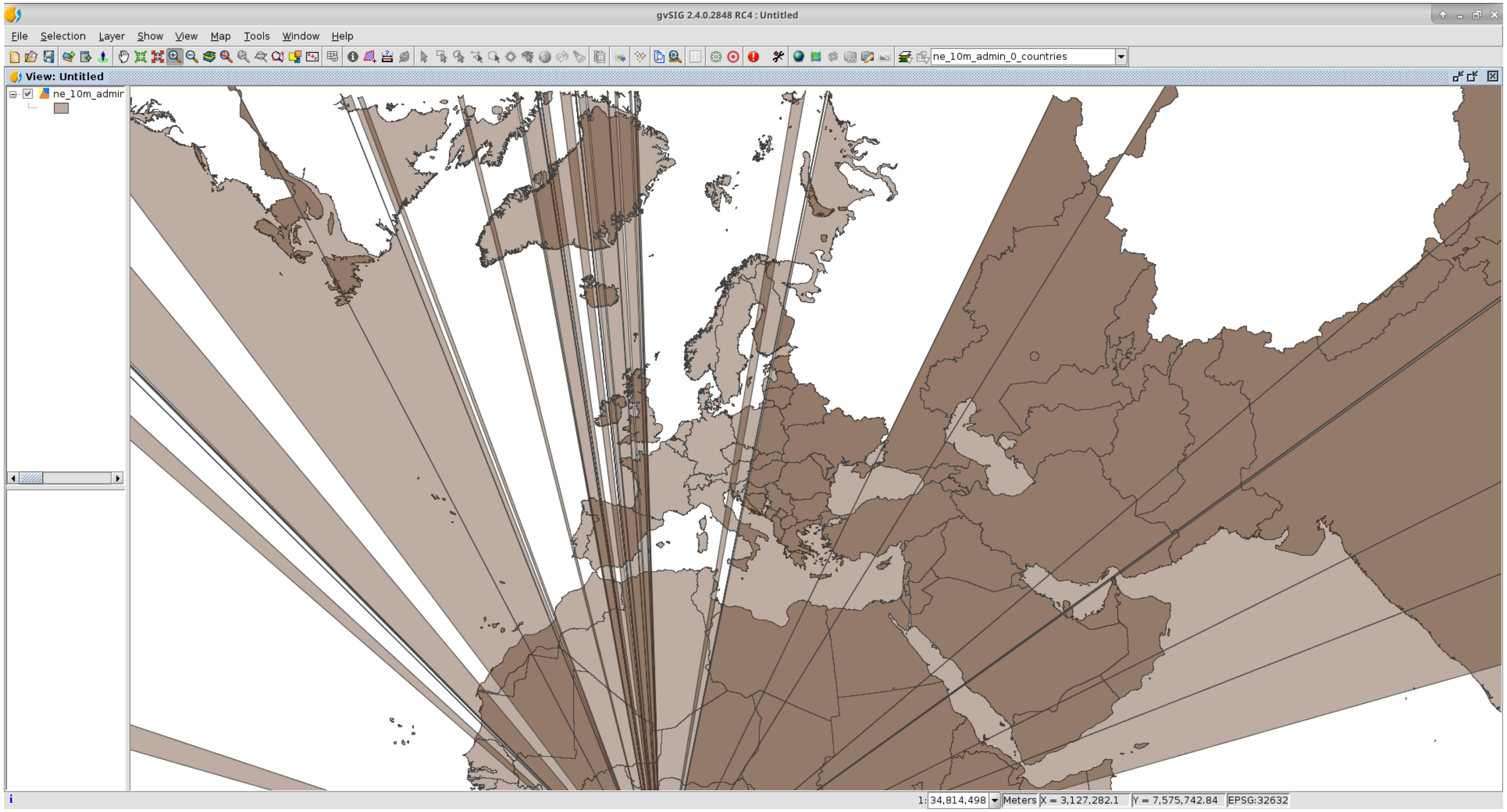
Remember: a CRS has an area of validity. On small portions one can transform between lat/long and UTM32N without problems. Ex. a shapefile in lat/long EPSG:4326 properly shows Italy in the EPSG:32632 projection:



…but once you get out of the area of validity:



And the same goes for gvSIG:



SCRIPTING

SCRIPTING IN GIS

Scripting is one of the powerful things in GIS (and in general). **Non developers** have the possibility to create great batch processes through it.

Different GIS have different scripting languages. In the open source world around 2009 a project named Geoscript was born. The project aimed to add spatial capabilities to dynamic scripting languages (**Javascript, Python, Scala and Groovy**).

Sadly **QGIS and gvSIG** adopted their own scripting languages. So in this course we will have a look at **pyQGIS**, the python based scripting language for QGIS (even if it is all but elegant, simple to remember or well documented).

LET’S START.

INSTALL QGIS

For this course we will use **QGIS 3.34.4 'Prizren'**.

Open the browser on the QGIS download page:

```
https://qgis.org/en/site/forusers/download.html
```

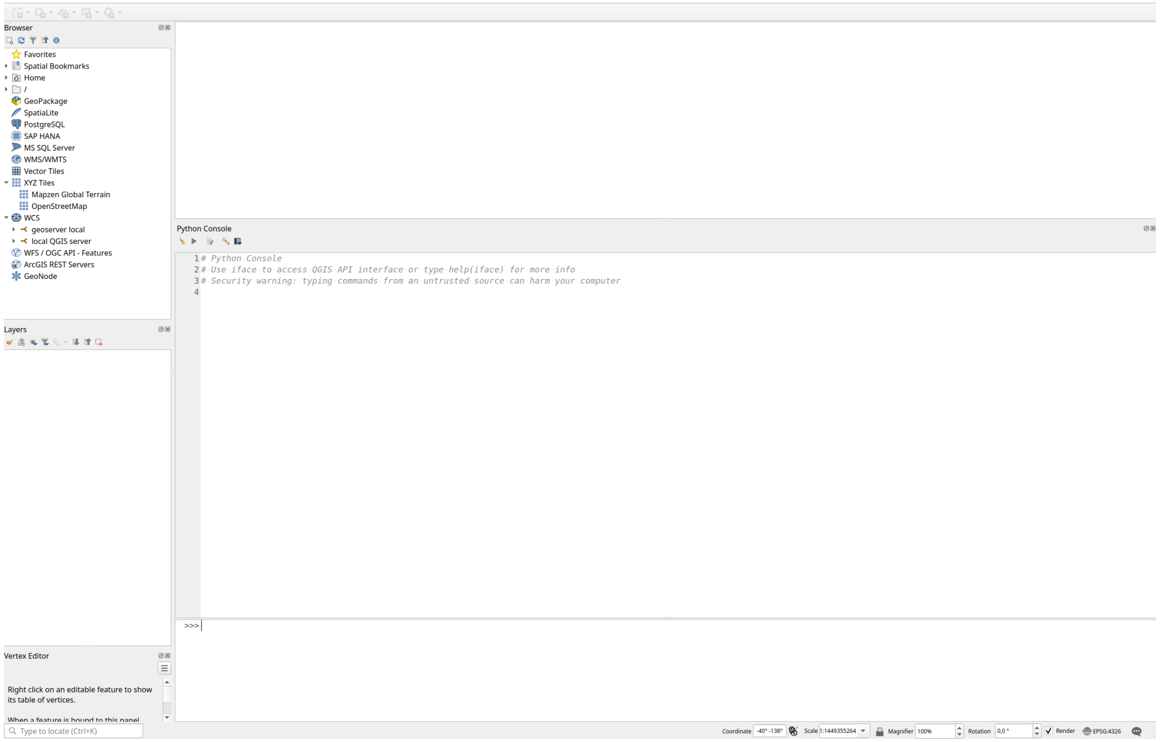
and download the installer for your operating system.

OPEN THE PYTHON CONSOLE

Open QGIS and create a new empty project. Then just open the Python Console from the menu:

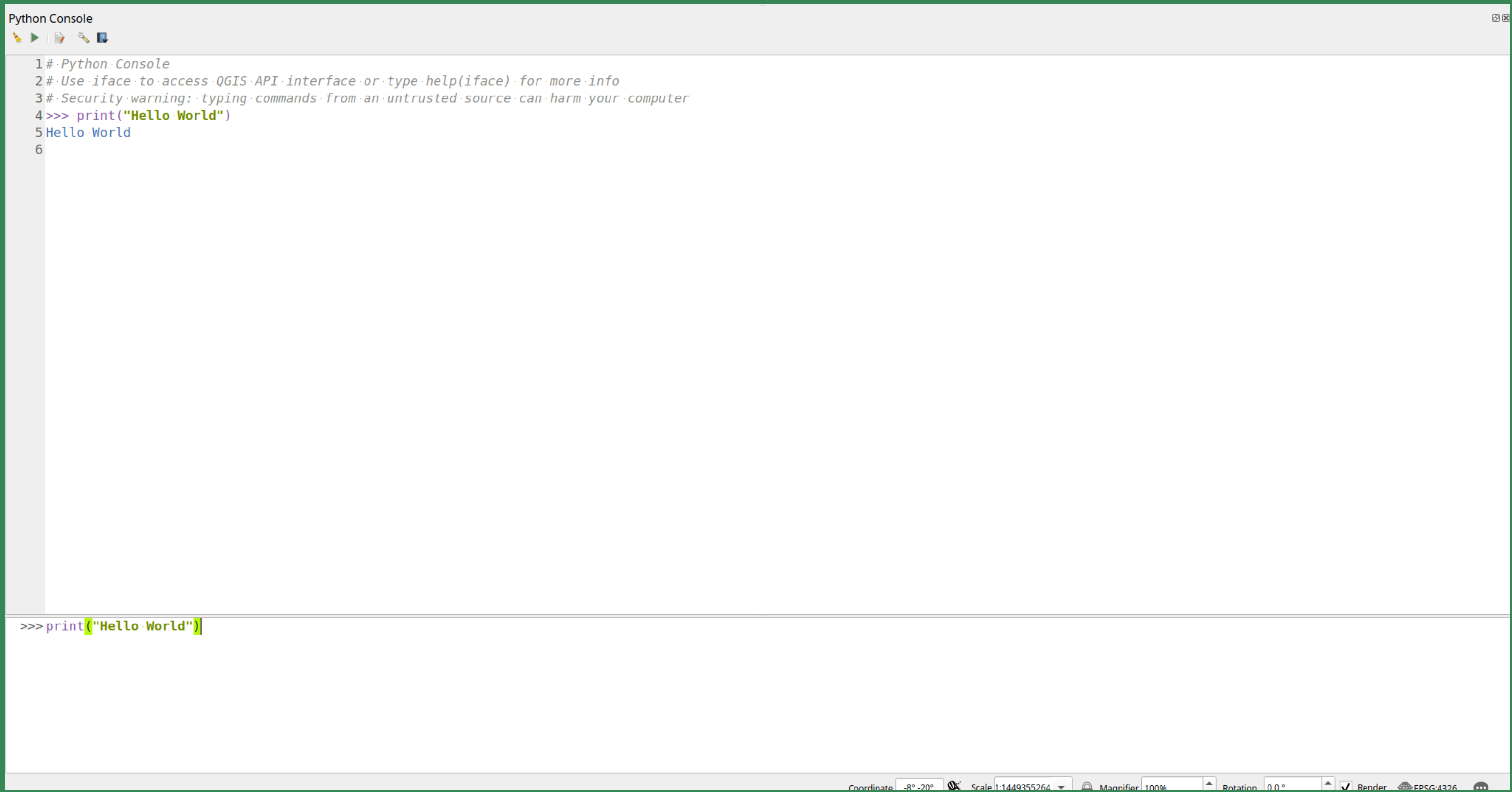
```
Plugins -> Python Console
```

The integrated Pyton Console should appear:



TEST THE CONSOLE

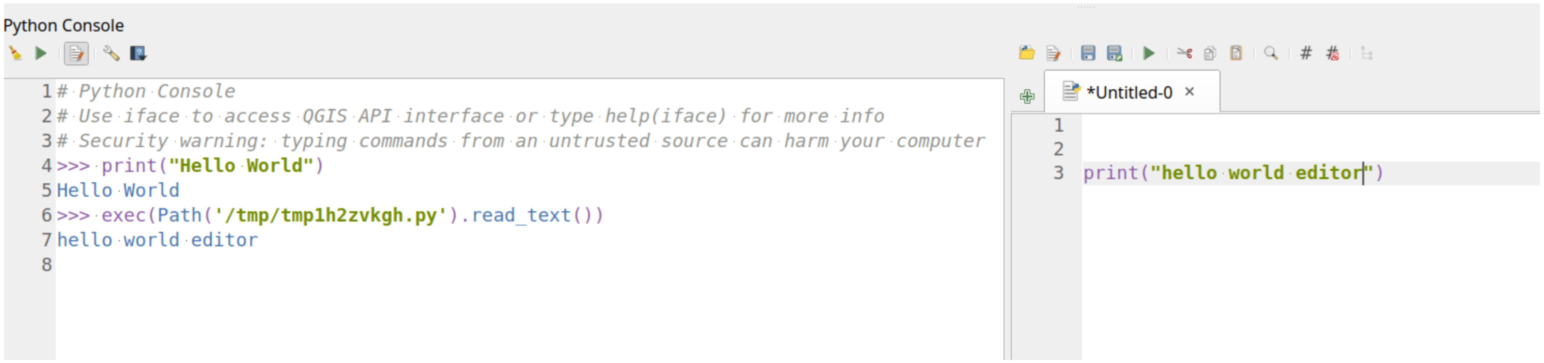
To make sure it works, test the Hello World python script:



and push the **Enter** button.

THE SCRIPTING EDITOR

Normally we do not write in the python console, we use a scripting editor, which can be opened from the 3rd icon:



Note that when you run the script (using the play button), in the console the script is executed calling exec on content of the script file:

```
exec(Path('/tmp/tmp1h2zvkgh.py').read_text())
```

and the output of the script (and its errors) are reported in the console.

```
<license>
  This work is released under Creative Commons Attribution
  Share Alike (CC-BY-SA).
</license>
<sources>
  Much of the knowledge needed to create this training material has
  been produced by the sparkling knights of the
  <a href="http://www.osgeo.org">OSGEO</a> and
  <a href="http://qgis.org">QGIS</a>,
  communities.
  Their websites are filled up with learning material that can be used
  to grow knowledge beyond the boundaries of this lessons
  Another essential source has been the Wikipedia project.
</sources>
<important>
  This work is part of the Advanced Geomatics Course given in 2023 at the
  EMMA Master of the Free University of Bolzano.
</important>
```