Challenge@PoliTO_**By Students**
Mapping PoliTO spaces – Kickoff meeting
6-7 March 2025

Mapping PoliTO spaces

Prof: Elisabetta Colucci, Emere Arco, Andrea Ajmar, Tommaso Calò, Simone Preziosi

Mentor: Alessandra Spreafico, Tommaso Calò, Simone Preziosi, Emere Arco

Click: Orazio Maria Pennisi, Laura Ronchetto

GIS INTRODUCTION

Challenge aims

The challenge focuses on exploring innovative solutions for **indoor space** mapping, aiming to enable tools for **indoor navigation**. Possible development directions include:

- Creating an **indoor navigation app**, applicable to the spaces of the Politecnico but extendable to other structures.
- Managing **cartographic data,** which can be modelled on specific DBMS or integrated into OpenStreetMap.
- Focusing on **indoor mapping projects**, with particular attention to building modelling to optimize indoor navigation.

The *geomatics* concept

- "discipline concerned with the collection, distribution, storage, analysis, processing, presentation of geographic data or geographic information" (1)
- uses **sensors** to acquire spatial data
  - terrestrial (total stations, GPS sensors, laser scanners, etc.; meteo/climatic/environmental stations)
  - marine (vessels, buoys, etc.; temperature, quality, levels, etc.)
  - airborne (aerial, RPAS; optical, radar, LiDAR, thermal, etc.)
  - satellite-based (optical, radar, LiDAR, thermal, etc.)
- transforms geographic data into **information**
- to sum up, **geomatics** is the coupling of **geographic** related methods and techniques with **informatic** ones

(1) ISO/TR 19122:2004(en) Geographic information/Geomatics — Qualification and certification of personnel

Geographic Information System

A **geographic information system** or **GIS** is a configuration of computer hardware and software specifically designed for the **acquisition, maintenance, and use** of cartographic data[1]

A Geographic Information Systems is a computer-based tool that examines **spatial relationships, patterns, and trends** in geography[2]

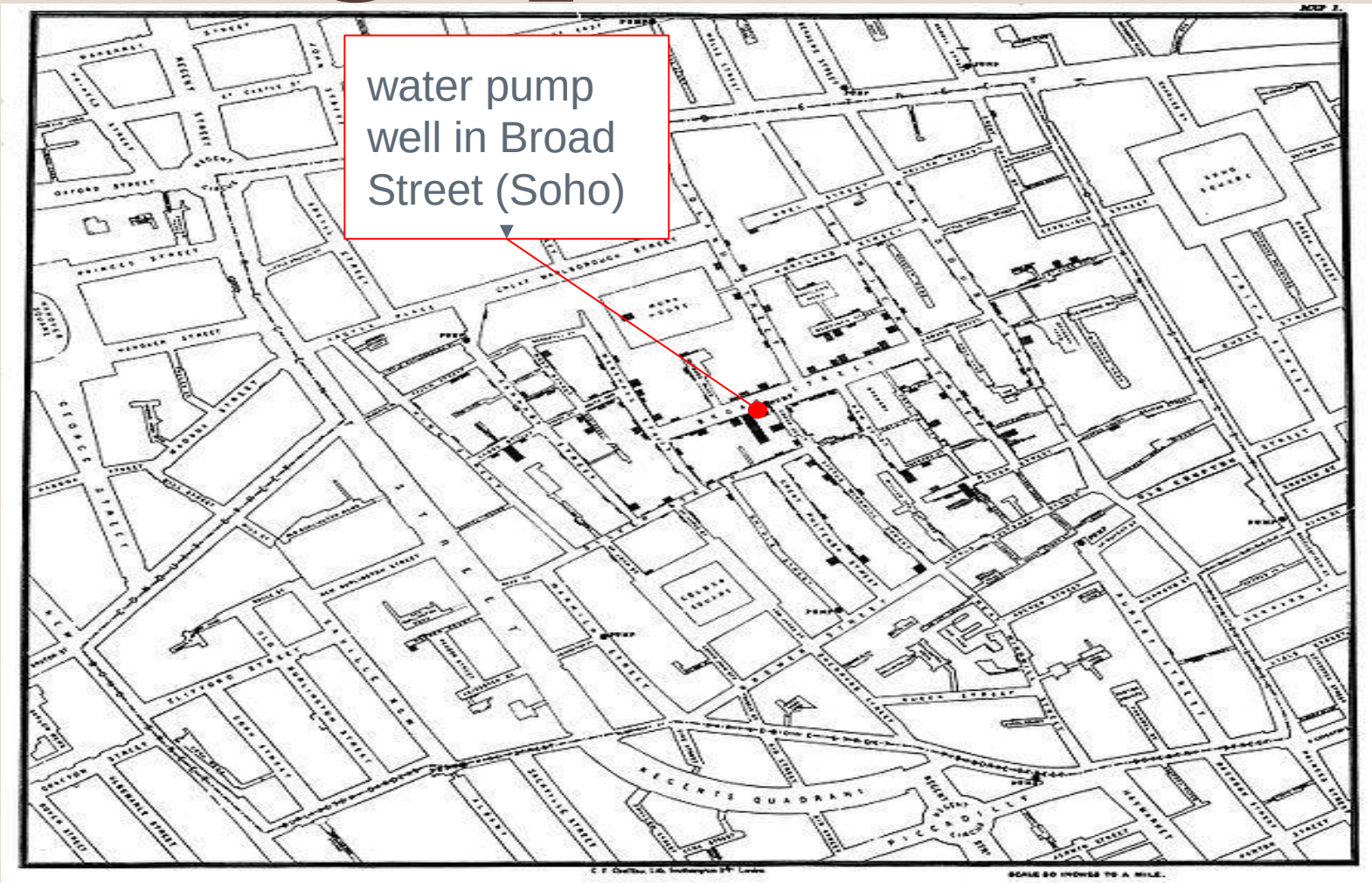
In a nutshell:

- Data without spatial reference doesn't provide geographic context. And without geographic context, you can't fully understand the world that we live in today
- That's why we need GIS and how come it's making a substantial impact in our daily lives

(1) Tomlins D. (1990) Geographic Information Systems and Cartographic Modeling. Englewood Cliffs, NJ: Prentice-Hall

(2) GISGeography

Geographic Information System



Original map by John Snow showing the **clusters of cholera cases** in the London epidemic of 1854

He marked the deaths from cholera, contributing to the scientific discovery that the disease was carried by water and not by air

- One of the earliest and most convincing demonstrations of the power and usefulness of thematic maps
- a forerunner to the modern GIS overlay analysis

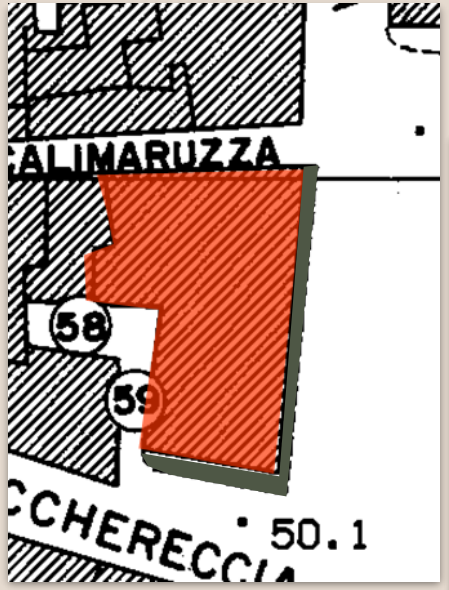
The map is somehow debated

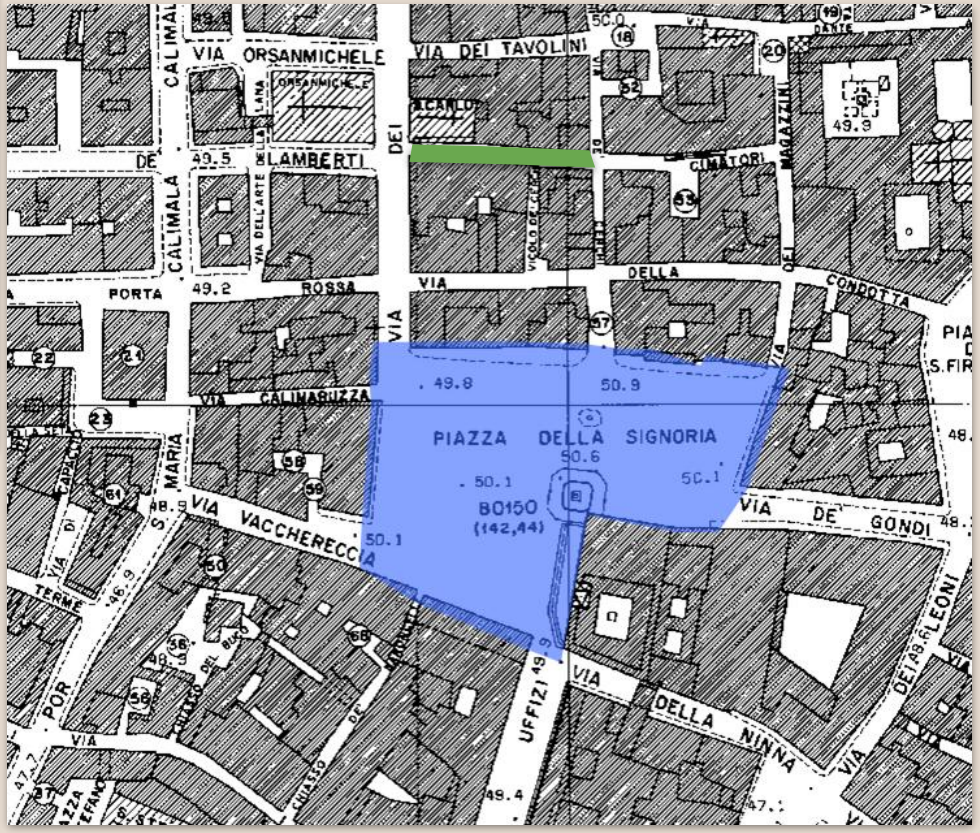
Extracted from Cartography by Kenneth Field

From traditional to digital cartography

The user of a **traditional cartography** scans for all entities that are reported (e.g. recognizes a house, a sidewalk, etc.)

However the power of analysis also allows him to gather other information not directly drawn (e.g. a square as a white space within buildings, a road trunk as an alonged space bordered by two sides by buildings, etc)



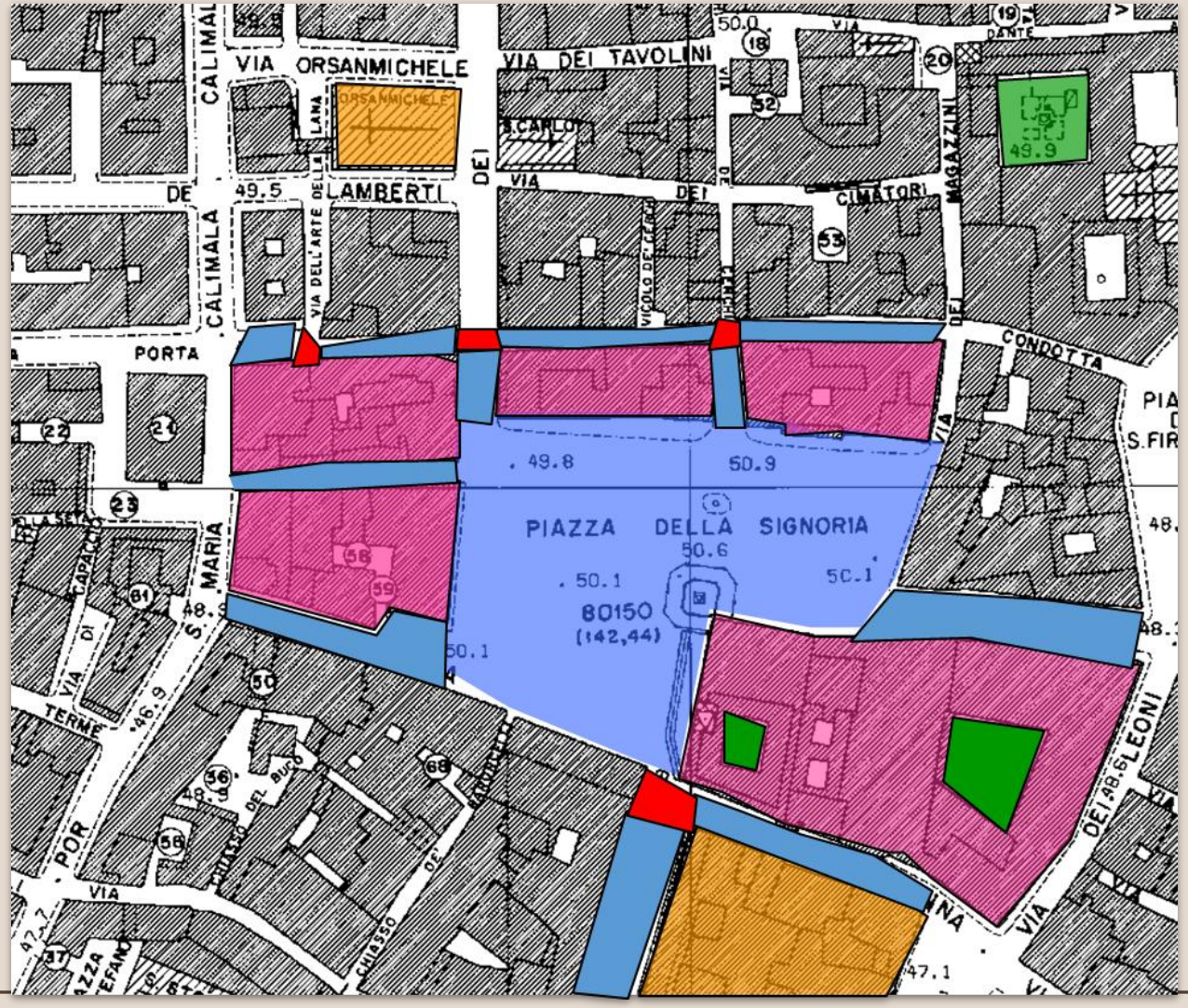


Courtesy of prof. Andrea Lingua, Politecnico di Torino

From traditional to digital cartography

In digital cartography, reading is not left to human observation. All recognizable entities (including “white spaces) must be defined (subject), described (by means of a geometrical shape and related attributes) and represented

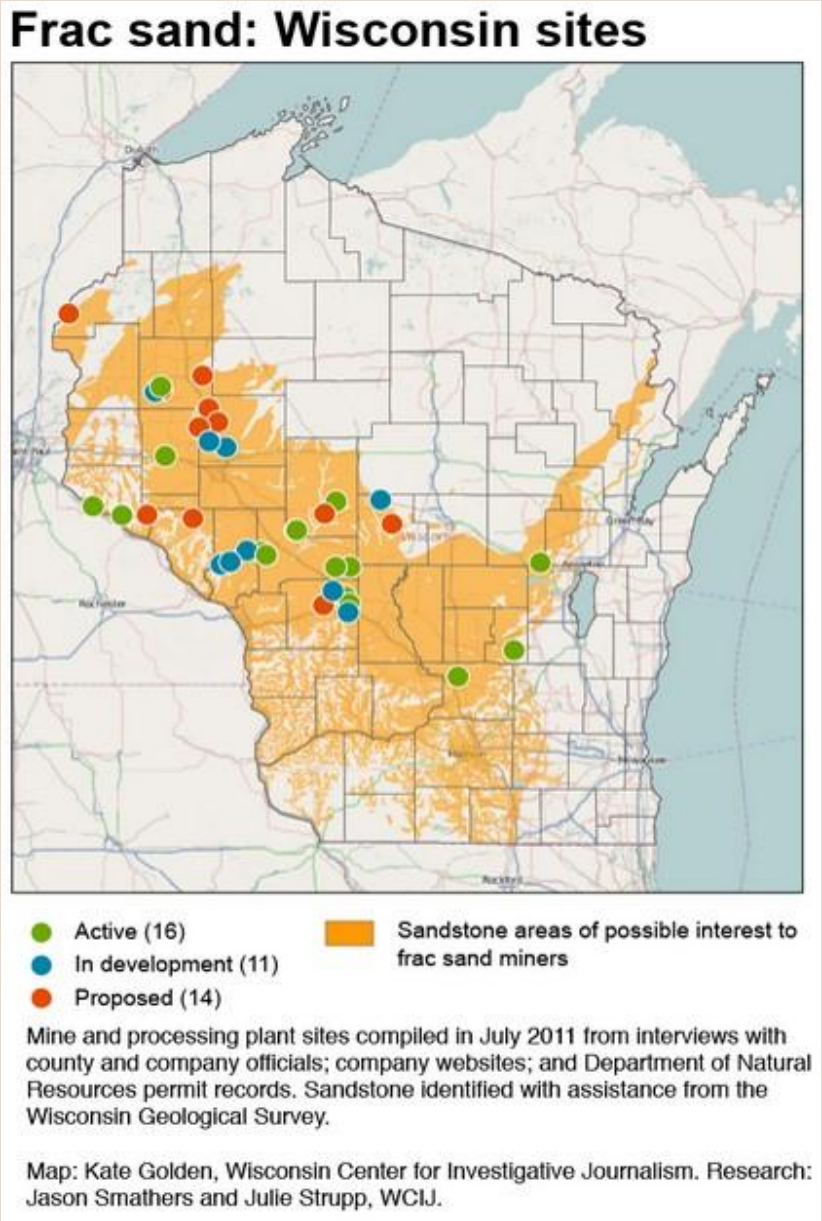
Example of entities are buildings, squares, road trunks, blocks, crossings, green areas, etc.



Courtesy of prof. Andrea Lingua, Politecnico di Torino

What can we do with GIS

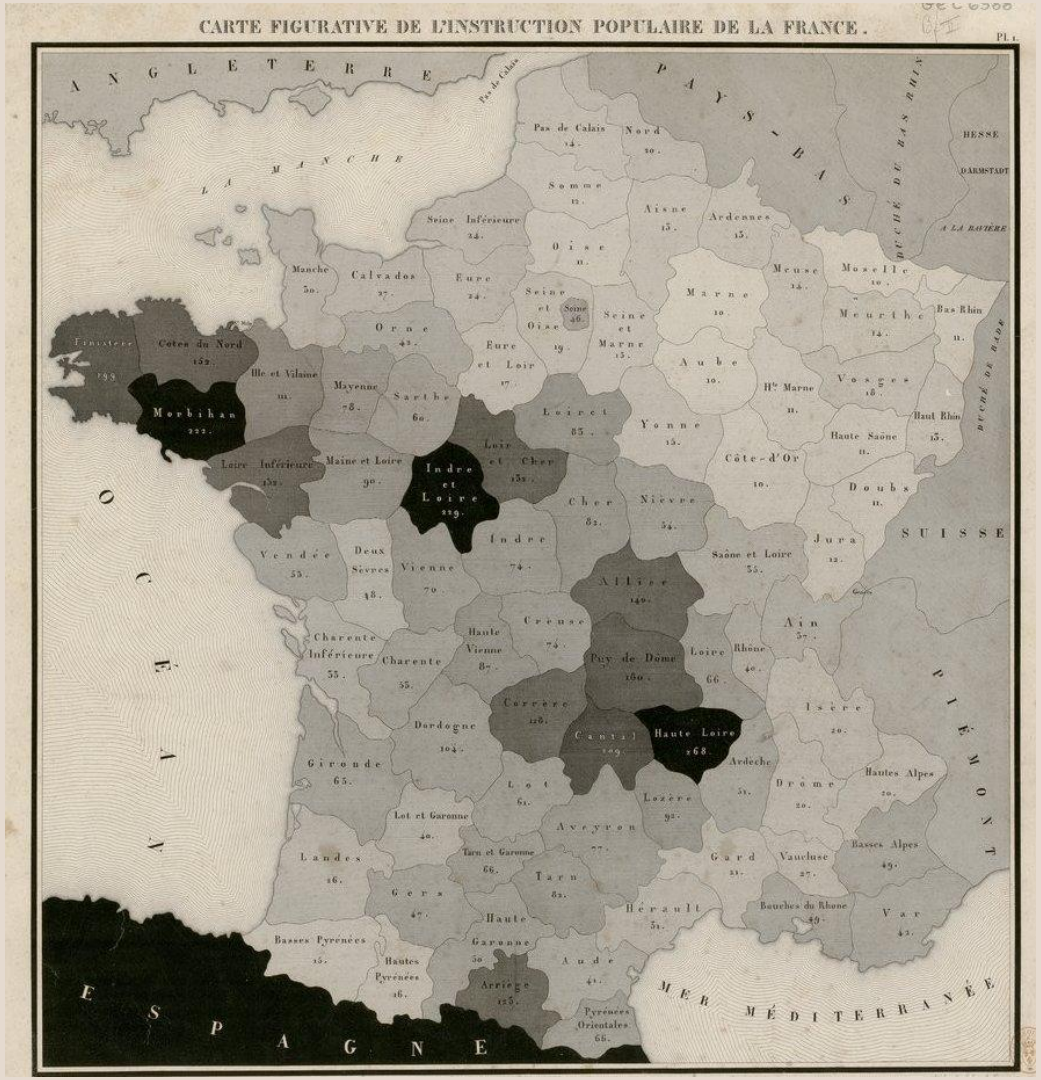
1. Mapping where things are



https://researchguides.library.wisc.edu/GIS

What can we do with GIS

1. Mapping where things are
2. Mapping quantities

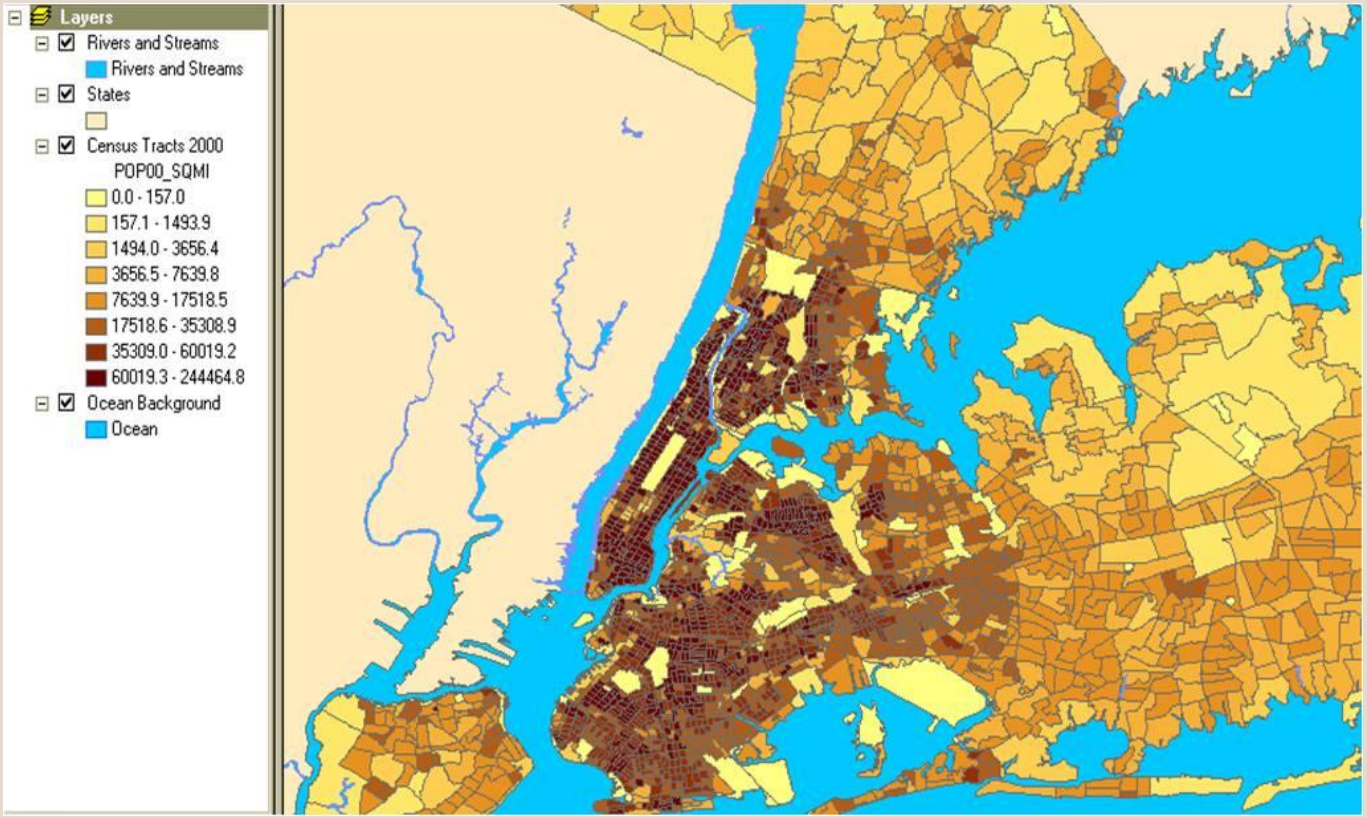


Dupin map of literacy in France (1826) is considered the first ever choropleth map

Legend: From 1 pupil per 10 inhabitants to 1 pupil per 268 inhabitants from lightest to darkest

What can we do with GIS

1. Mapping where things are
2. Mapping quantities
3. Mapping densities



https://researchguides.library.wisc.edu/GIS

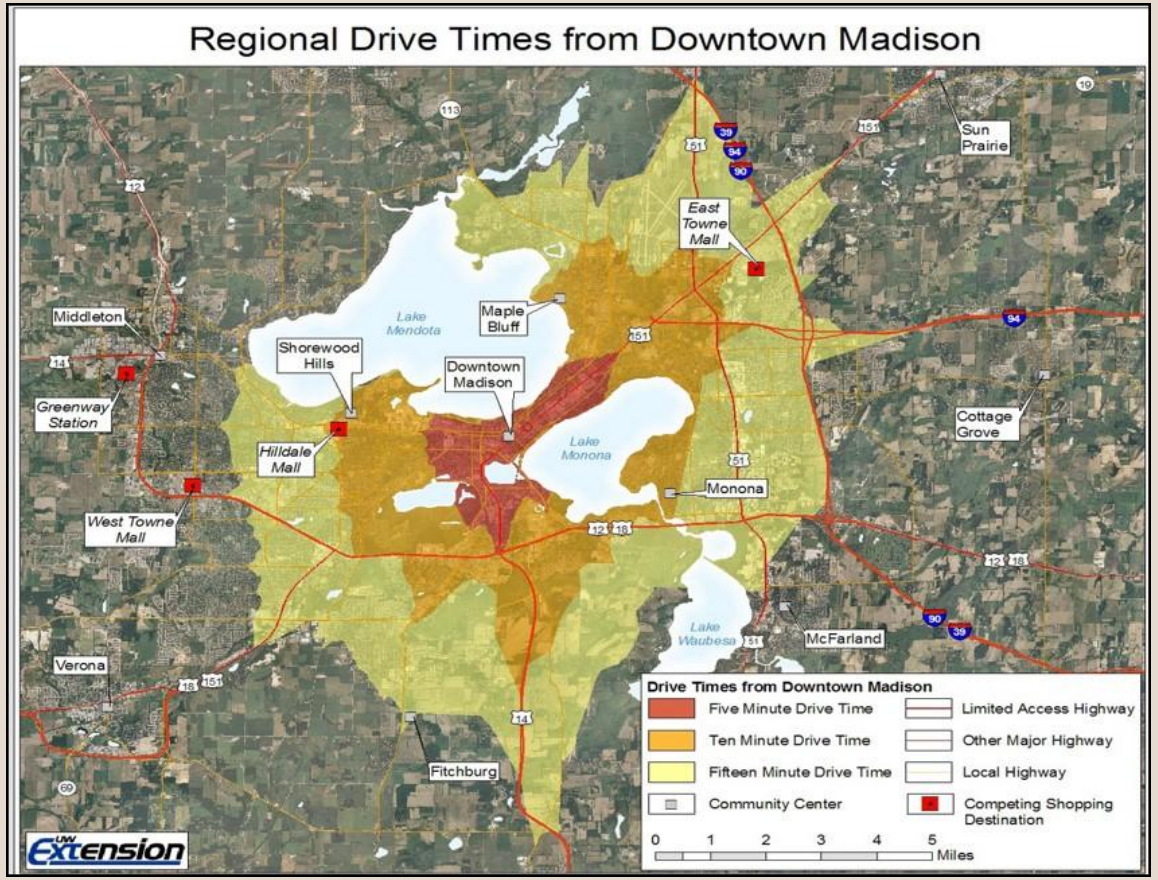
What can we do with GIS

1. Mapping where things are
2. Mapping quantities
3. Mapping densities
4. Finding what is inside

https://researchguides.library.wisc.edu/GIS

What can we do with GIS

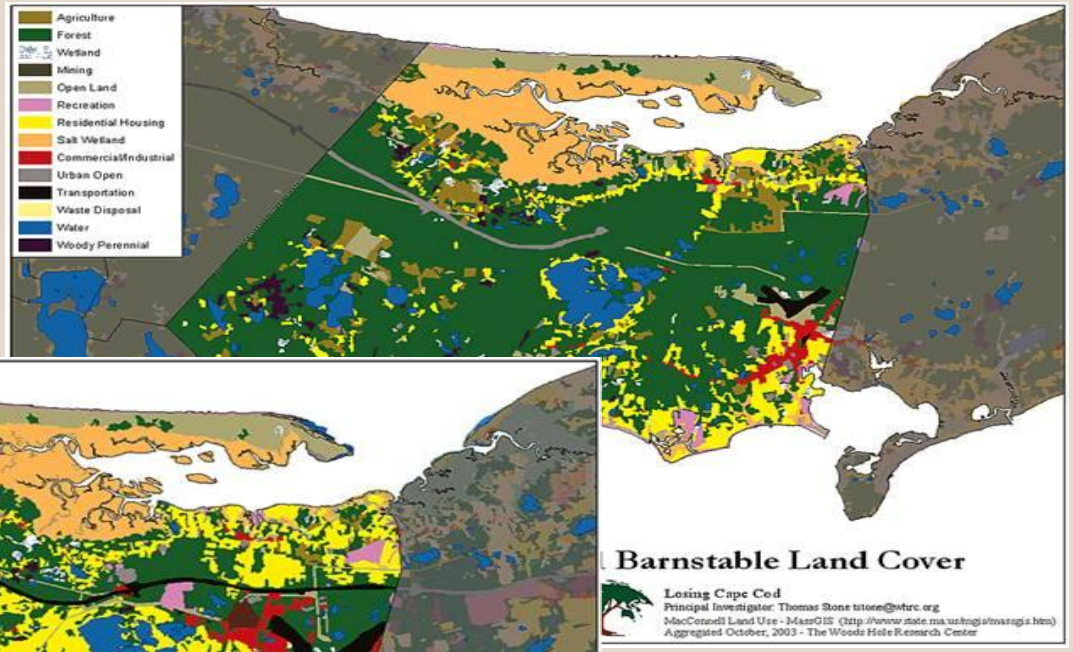
1. Mapping where things are
2. Mapping quantities
3. Mapping densities
4. Finding what is inside
5. Finding what is nearby

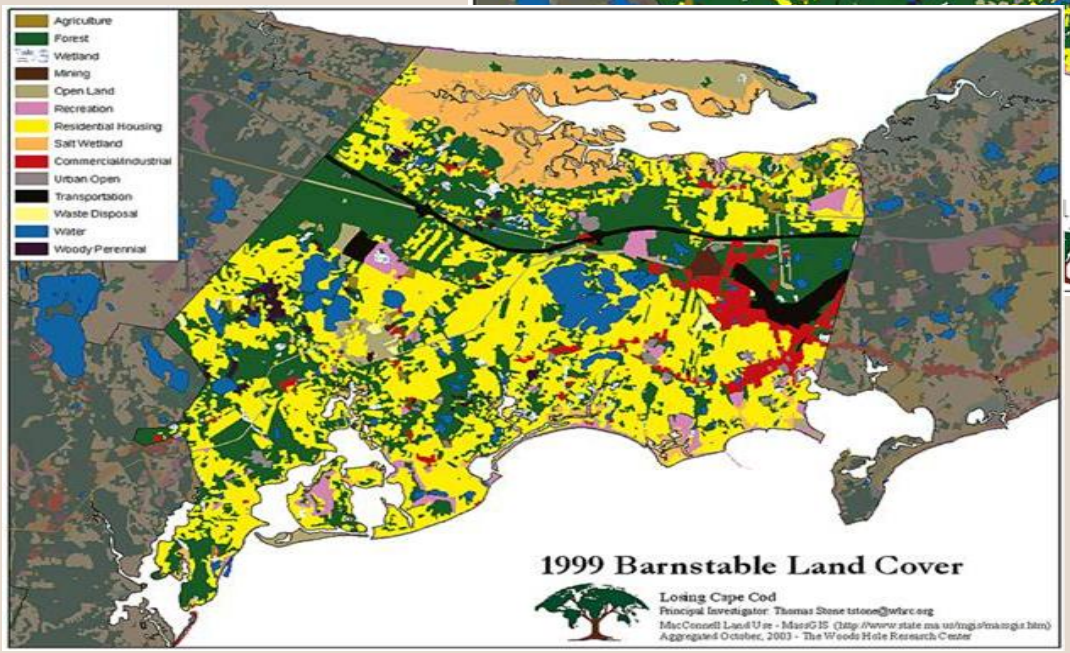


https://researchguides.library.wisc.edu/GIS

What can we do with GIS

1. Mapping where things are
2. Mapping quantities
3. Mapping densities
4. Finding what is inside
5. Finding what is nearby
6. Mapping change
7. ...





https://researchguides.library.wisc.edu/GIS

Computer-Aided Design

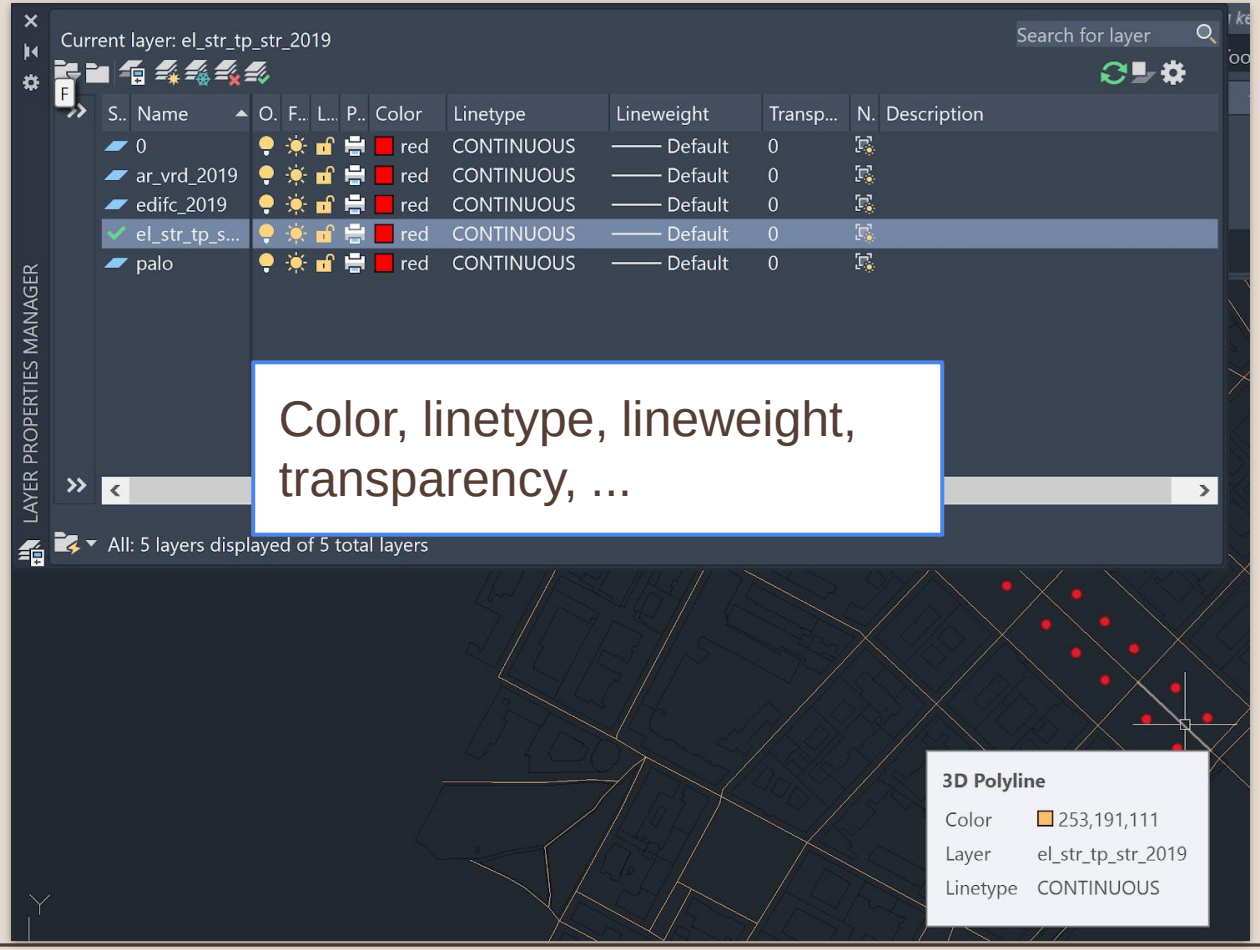
"Computer-aided design (**CAD**) is the use of computers (or workstations) to aid in the creation, modification, analysis, or optimization of a **design**" (1)

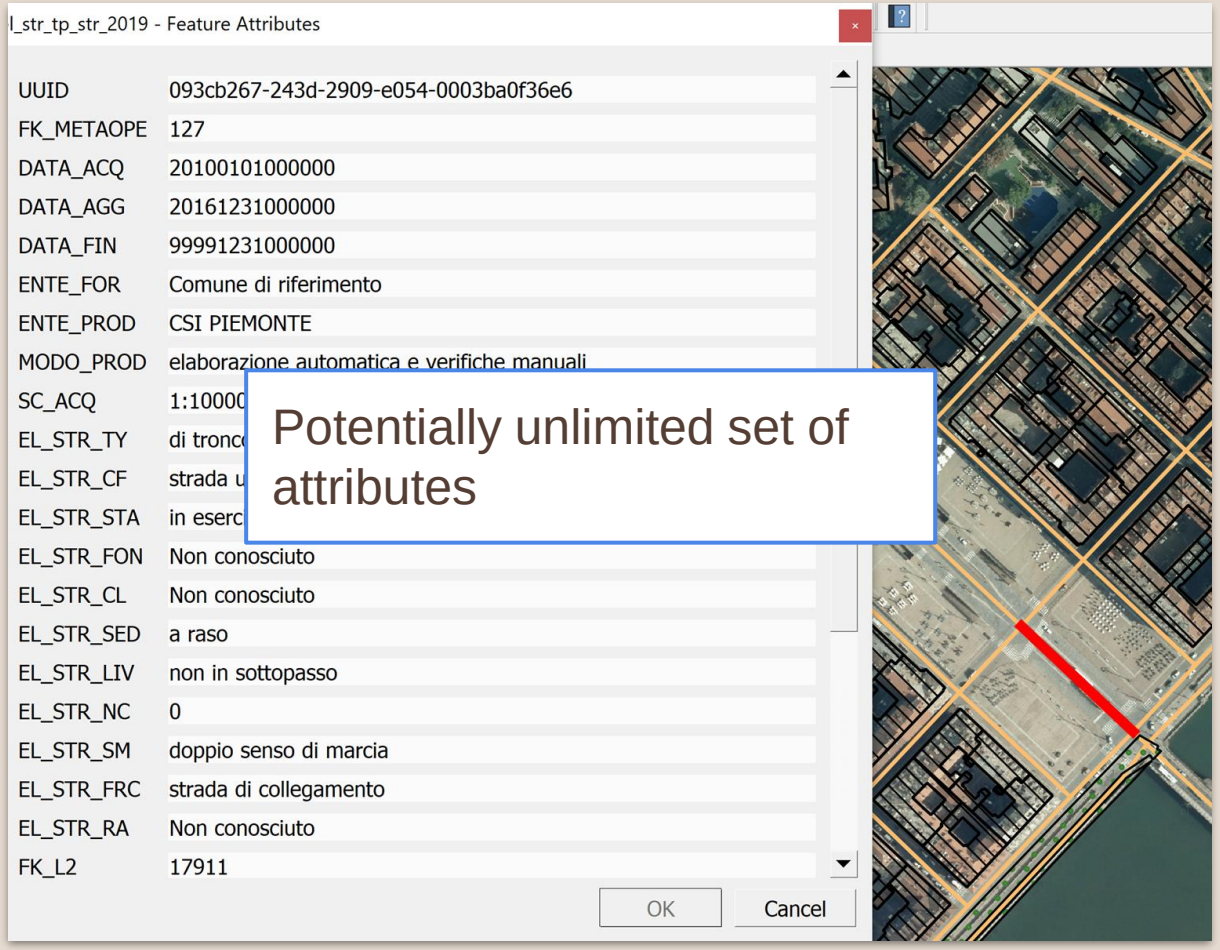
(1) Narayan, K. Lalit (2008). Computer Aided Design and Manufacturing. New Delhi: Prentice Hall of India. p. 3. ISBN 978-8120333420

CAD vs. GIS

CAD	GIS
an object is described by means of its **graphic representation**	an object has a shape and a set of attributes (**database**)
coordinates are referred to an internal 2D/3D reference system, **not considering local deformations**	coordinates are referred to a specific reference system (**projection + datum**)
big scales	**multiscale**
focus on **drawing functions**	simplified geometric primitives and advanced **analysis instruments**

CAD vs. GIS


Current layer: el_str_tp_str_2019
Search for layer
LAYER PROPERTIES MANAGER
S.. Name O.. F.. L... P.. Color Linetype Lineweight Transp... N.. Description
0 red CONTINUOUS Default 0
ar_vrd_2019 red CONTINUOUS Default 0
edifc_2019 red CONTINUOUS Default 0
el_str_tp_s... red CONTINUOUS Default 0
palo red CONTINUOUS Default 0
All: 5 layers displayed of 5 total layers
Color, linetype, lineweight, transparency, ...
3D Polyline
Color 253,191,111
Layer el_str_tp_str_2019
Linetype CONTINUOUS


l_str_tp_str_2019 - Feature Attributes
UUID 093cb267-243d-2909-e054-0003ba0f36e6
FK_METAOPE 127
DATA_ACQ 20100101000000
DATA_AGG 20161231000000
DATA_FIN 99991231000000
ENTE_FOR Comune di riferimento
ENTE_PROD CSI PIEMONTE
MODO_PROD elaborazione automatica e verifiche manuali
SC_ACQ 1:10000
EL_STR_TY di tronc
EL_STR_CF strada u
EL_STR_STA in eserc
EL_STR_FON Non conosciuto
EL_STR_CL Non conosciuto
EL_STR_SED a raso
EL_STR_LIV non in sottopasso
EL_STR_NC 0
EL_STR_SM doppio senso di marcia
EL_STR_FRC strada di collegamento
EL_STR_RA Non conosciuto
FK_L2 17911
OK
Cancel
Potentially unlimited set of attributes

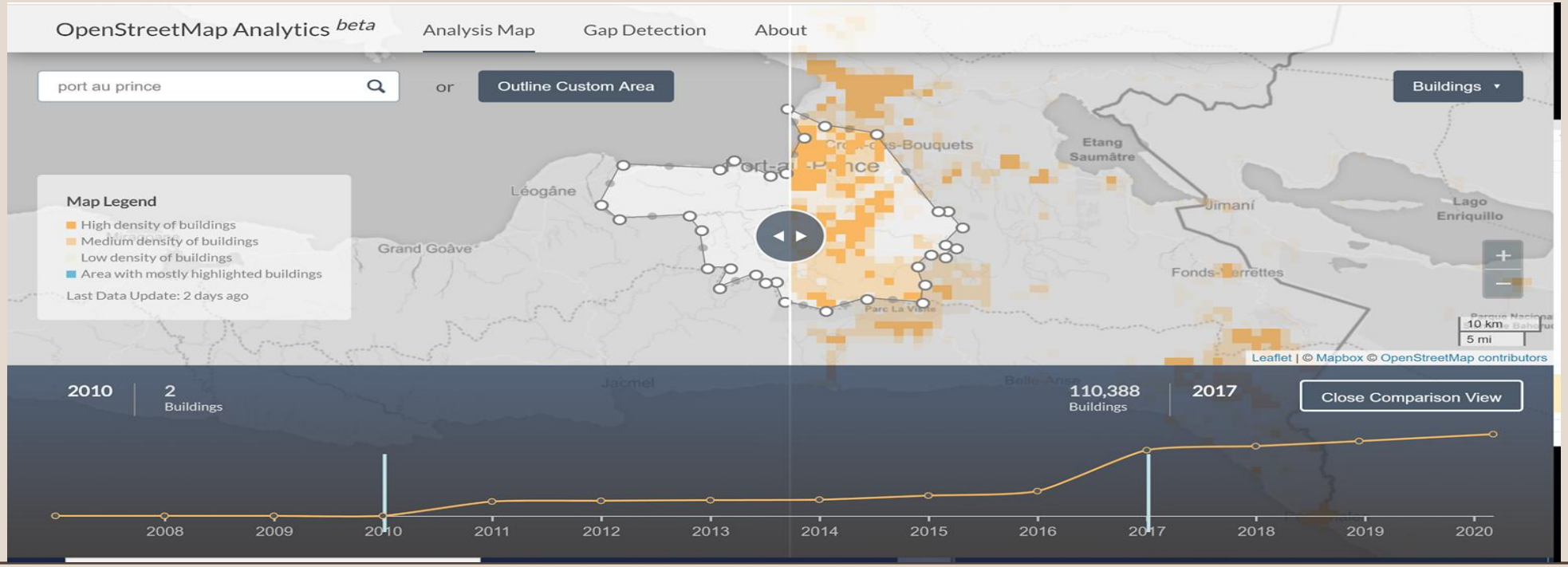
CAD vs. GIS

A **GIS** system can be considered as an **evolution of a CAD environment**, as:

- it is based on CAD technologies for the geometric design component (simplifying the geometric primitives), **extending the descriptive capabilities** of an object and the ability **to manage significant amounts of data** by relying on database models
- constitutes a **more flexible format**, in which the same data can be used easily for different purposes (i.e. the realization of different symbolizations of the same data is simplified)

Volunteered geographic information

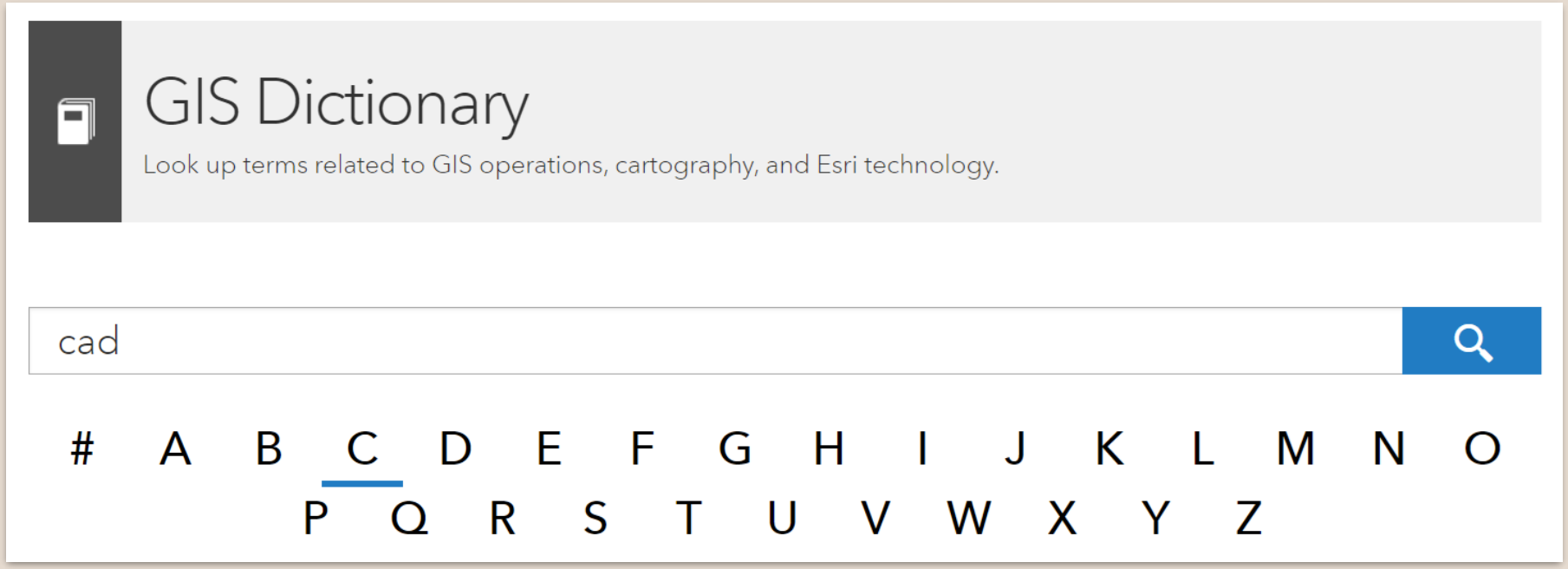
Is the harnessing of tools to create, assemble, and disseminate geographic data provided voluntarily by individuals. Most relevant examples are OpenStreetMap and WikiMapia (differs on licensing conditions)



Port-au-Prince earthquake

Hurricane Matthew

A GIS Dictionary



https://support.esri.com/en/other-resources/gis-dictionary

GIS Software

GIS software categories

- Desktop GIS: the most common user interface to GIS data
- Spatial DBMS: database management systems with dedicated geospatial modules (e.g., Oracle/Oracle Spatial, PostgreSQL/PostGIS)
- Web map server: server side components used to share and edit geographic data on the Web (e.g., ArcGIS Server, GeoServer)
- Software development frameworks (desktop): processing librairies (e.g. GDAL, Orfeo Toolbox)

(Incomplete) list of desktop GIS software solutions

Commercial off-the-shelf (COTS)

- ESRI (ArcGIS Pro, ArcMAP)
- Intergraph (GeoMedia)
- Autodesk (interfaces with AutoCAD)
- ENVI (remote sensing)
- ERDAS IMAGINE (remote sensing)

Open source (OS)

- QGIS
- GRASS GIS (modules integrated in QGIS)
- SAGA GIS
- ILWIS (remote sensing)

GIS software comparison

License, source, & operating system support [edit]

GIS software	Free software	Open source	Windows	Mac OS X	Linux	BSD	Unix	Web	Other
ArcGIS	Viewer(s)	No	Yes	No[1]	No	No	Yes	Yes	Google Earth Plugin, KML, WMS
Autodesk	Viewer(s)	No	Yes	No	Yes	No	No	Yes	No
Cadcorp	Viewer(s)	No	Yes	No	No	No	No	Yes	No
CAPAWARE	Yes	Yes	Yes	No	No	No	No	No	No
Deegree	Yes	Yes	Java	Java	Java	Java	Java	Yes	No
Erdas Imagine	Viewers & Plug-ins	No	Yes	No	No	No	No	Yes	No
GeoBase - Telogis	Trial	No	Yes	No	Yes	Yes	No	Yes	Traffic, WMS, ADAS, Routing
GeoNetwork	Yes	Yes	Java	Java	Java	Java	Java	Yes	No
GeoServer	Yes	Yes	Java	Java	Java	Java	Java	Java	No
GeoTools	Yes	Yes	Java	Java	Java	Java	Java	No	No
GRASS	Yes	Yes	Yes	Yes	Yes	Yes	Yes	via pyWPS[2]	No
gvSIG	Yes	Yes	Java	Java	Java	Java	Java	No	No
IDRISI	No	No	Yes	No	No	No	No	No	No
ILWIS	Yes	Yes	Yes	No	No	No	No	No	No
GeoMedia	Viewer(s)	No	Yes	No	No	No	CLIX	Yes	KML
JUMP GIS	Yes	Yes	Java	Java	Java	Java	Java	No	No
Kosmo	Yes	Yes	Java	Java	Java	Java	Java	No	No
LandSerf	No	No	Java	Java	Java	Java	Java	No	No
MapDotNet	No	No	Yes	No	No	No	No	Yes	
Manifold System	No	No	Yes	No	No	No	No	Yes	No
Microsoft MapPoint	Discontinued	No	Yes	No	No	No	No	Yes	No
Pitney Bowes MapInfo Pro	Viewer(s)	No	Yes	No	No	No	Yes	Yes	No
MapServer	Yes	Yes	Yes	Yes	Yes	Yes	Yes	AMP	No
Maptitude Caliper software	No	No	Yes	No	No	No	No	Yes	No
MapWindow GIS	Yes	Yes	Yes	No	No	No	No	No	No
Oracle Spatial	No	No	Yes	Yes	Yes	No	Yes	Yes	No
PostGIS	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
QGIS	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Google Earth Plugin, KML, WMS
RegioGraph	No	No	Yes	No	No	No	No	No	No
RemoteView	No	No	Yes	No	No	No	No	No	No
SAGA GIS	Yes	Yes	Yes	Yes	Yes	Yes	No	No	KLM
SAP HANA	Free Trial	No	No	No	Yes	No	No	Yes	No
Smallworld	No	Yes	Yes	No	Yes	No	Yes	Read-only	OLE, XML, Multiuser, Utility Data Model, Data Translator export/import
SPRING	Yes	Yes	Yes	No	Yes	No	Solaris	No	No
TerraLib TerraView	Yes	Yes	Yes	No	Yes	No	No	No	No
TNTmips	Viewer(s)	No	Yes	Yes	Yes	No	Yes	No	No
TransModeler Caliper Software	No	No	Yes	No	No	No	No	No	No
uDIG	Yes	Yes	Java	Java	Java	Java	Java	No	No

GIS DATA TYPES

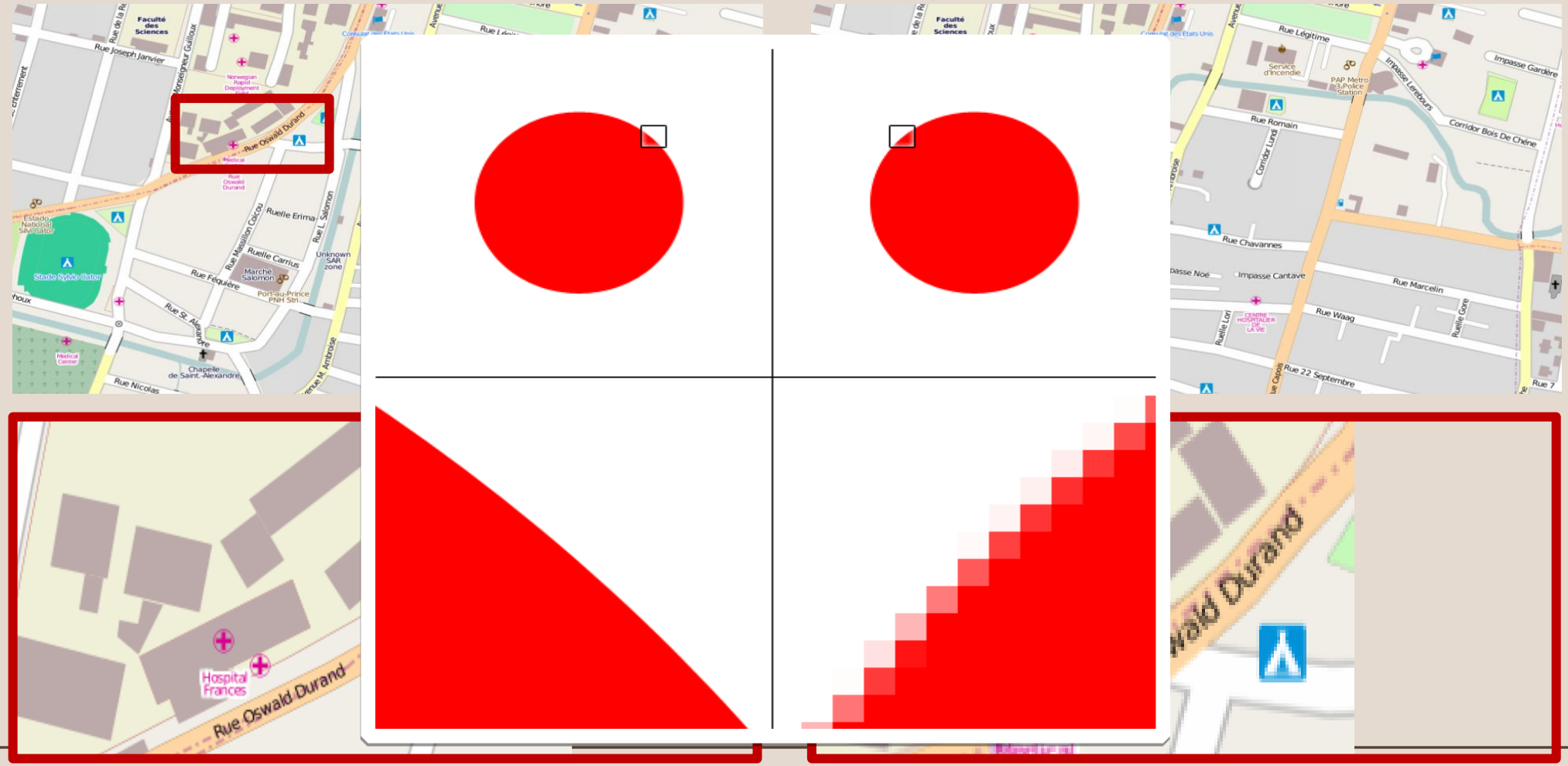
GIS data types

Vector Data vs Raster Data

Courtesy of prof. Fabio Giulio Tonolo, Politecnico di Torino

GIS data types

Vector Data vs Raster Data

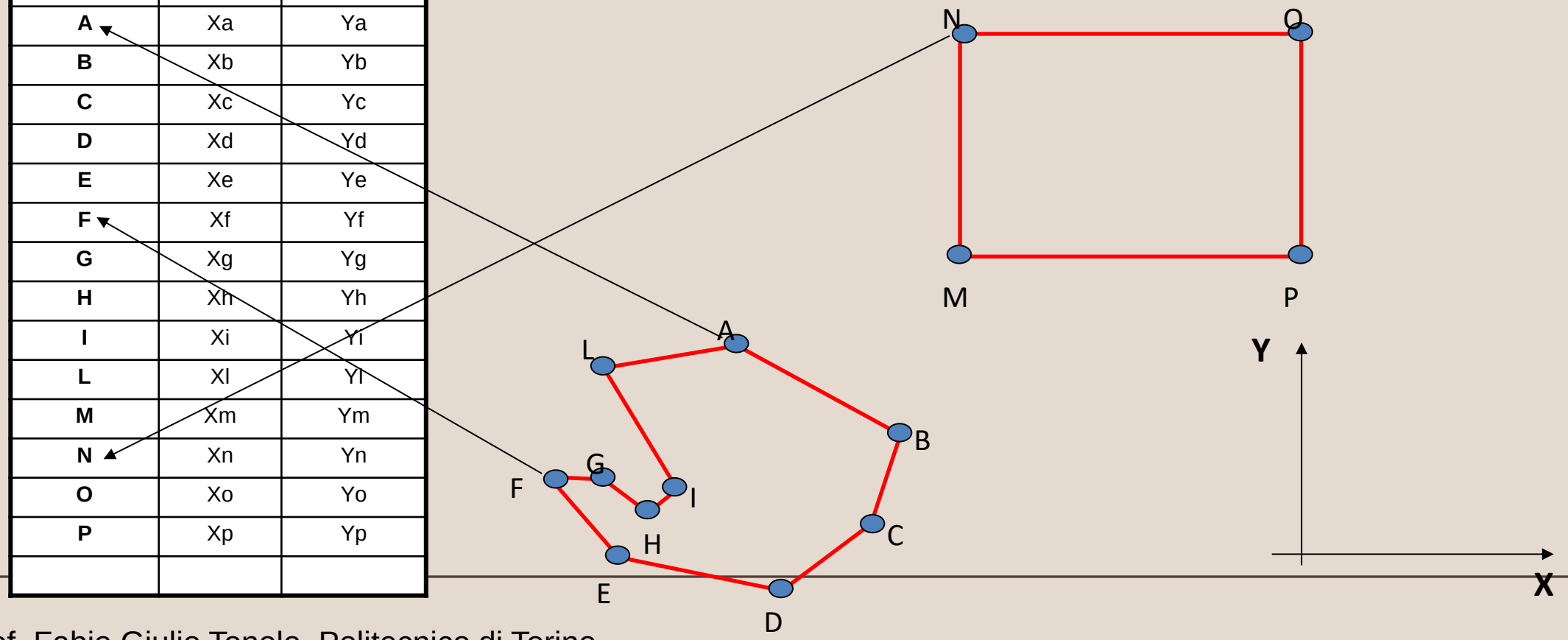


Courtesy of prof. Fabio Giulio Tonolo, Politecnico di Torino

Vector Data

Vector data are made of **features** like points, lines and polygons, coded and stored on the basis of their **coordinates** in respect to a well known coordinate system.

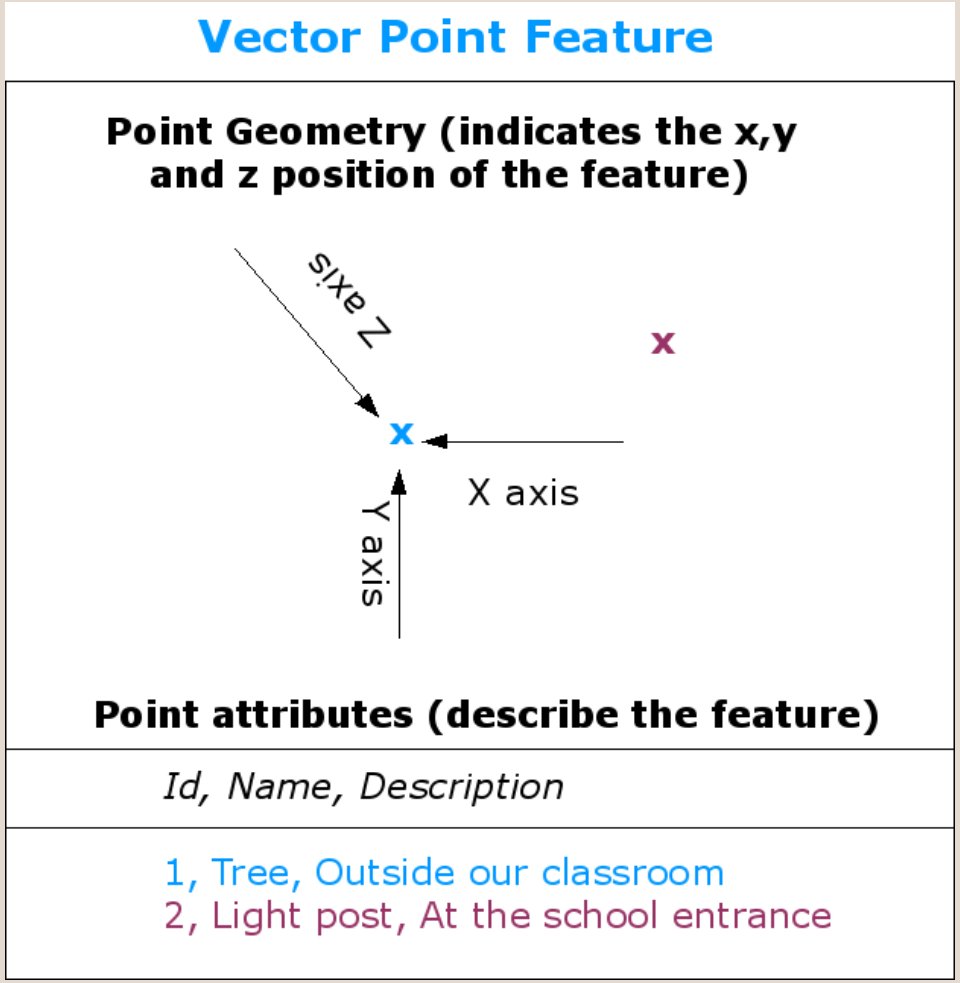
ID punto	X	Y
A	Xa	Ya
B	Xb	Yb
C	Xc	Yc
D	Xd	Yd
E	Xe	Ye
F	Xf	Yf
G	Xg	Yg
H	Xh	Yh
I	Xi	Yi
L	Xl	Yl
M	Xm	Ym
N	Xn	Yn
O	Xo	Yo
P	Xp	Yp



Courtesy of prof. Fabio Giulio Tonolo, Politecnico di Torino

Vector geometry types

Points - Zero-dimensional points are used for geographical features that can best be expressed by a single point reference. Points can also be used to represent areas when displayed at a small scale. A point feature is described by its X, Y and optionally Z coordinate



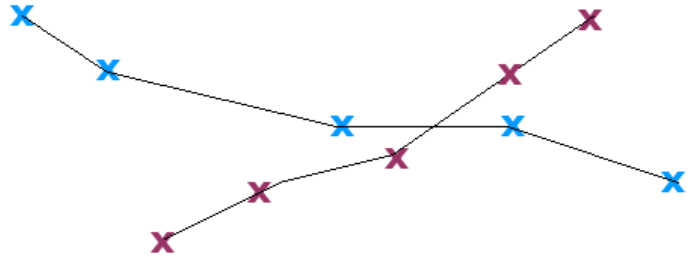
https://docs.qgis.org/2.8/en/docs/gentle_gis_introduction/vector_data.html#overview

Vector geometry types

Polylines - One-dimensional lines or polylines are used for linear features. Again, as with point features, linear features displayed at a small scale will be represented as linear features rather than as a polygon. A polyline is **a sequence of joined vertices**. Each vertex has an X, Y and optionally Z coordinate

Vector Polyline Feature

Polyline Geometry (a series of connected vertices that do not form an enclosed shape)



Polyline attributes (describe the feature)

Id, Name, Description
1, Footpath 1, From class to the playground 2, Footpath 2, From the school gate to the hall

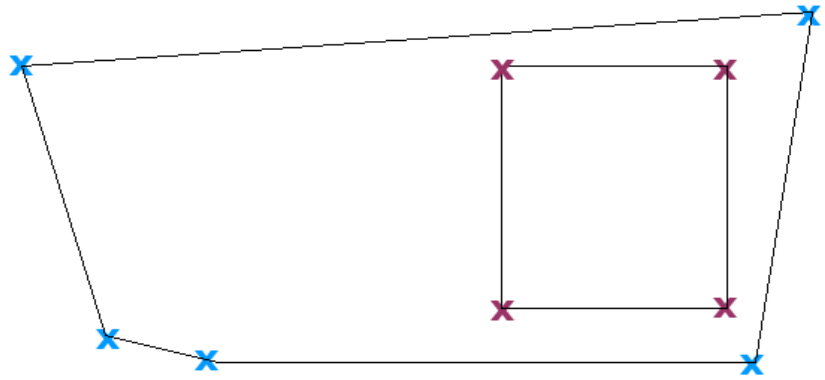
https://docs.qgis.org/2.8/en/docs/gentle_gis_introduction/vector_data.html#overview

Vector geometry types

Polygons - Two-dimensional polygons are used for geographical features that cover a particular area of the earth's surface. A polygon, like a polyline, is a **sequence of vertices**. However in a polygon, **the first and last vertices are always at the same position.**

Vector Polygon Feature

Polygon Geometry (a series of connected vertices that do form an enclosed shape)



Polygon attributes (describe the feature)

Id, Name, Description
1, School Boundary, Fenceline for the school 2, Sports Field, We play soccer here

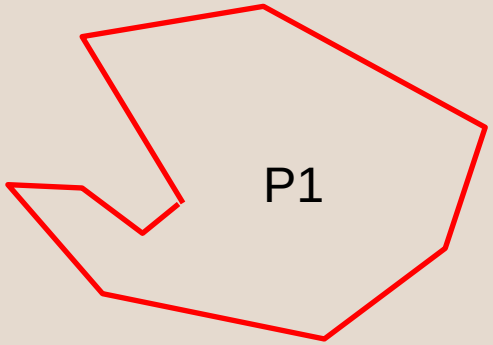
https://docs.qgis.org/2.8/en/docs/gentle_gis_introduction/vector_data.html#overview

Data descriptive of the features

Attribute table: an attribute is a characteristic of the geometric feature; it can be expressed in alphanumeric format (digits & letters)

Feature ID (FID)

Unique identifier of the feature the information belongs to



Attribute table

FID	nome	Area
P1	aaa	25
P2	bbb	12
P3	ccc	4

Field

Record

Value

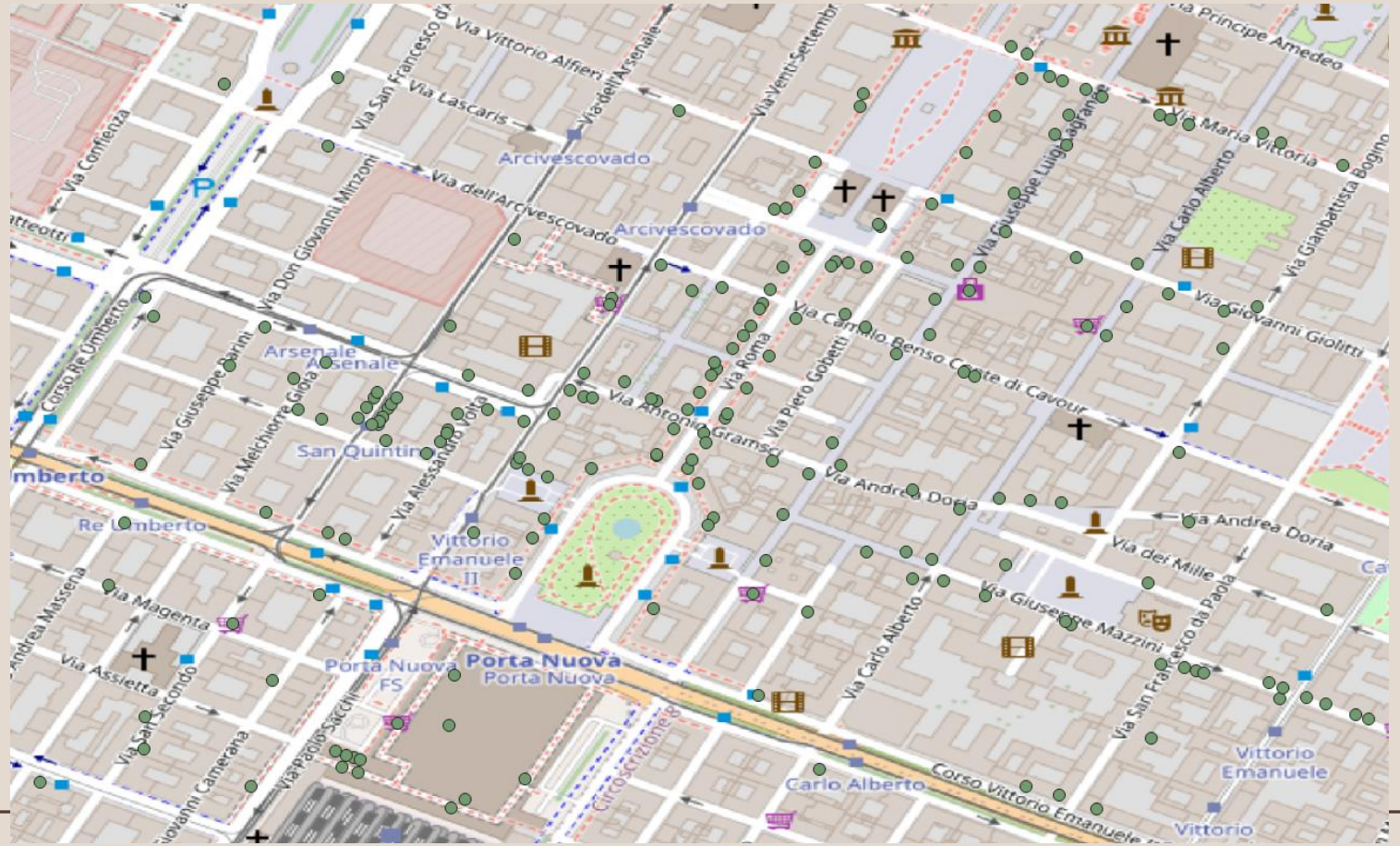
Vectors vs. Rasters

- **Raster** datasets record a value for all points in the area covered which may require **more storage** space than representing data in a vector format that can store data only where needed
- **Raster** data allows easy implementation of **overlay operations**, which are more difficult with vector data
- Vector data can be displayed as vector graphics used on traditional maps, whereas **raster** data will appear as an image that may have a **blocky appearance** for object boundaries (depending on the resolution of the raster file)
- **Vector** data is more compatible with **relational database** environments
- **Vector** data is simpler to **update and maintain**
- **Vector** data allows much more **analysis capability**, especially for "networks" such as roads, power, rail, telecommunications, etc.
- **Vector** data can be easier to **register, scale, and re-project**, which can simplify combining vector layers from different sources

Some examples

Point geometry

Green points represent shops position



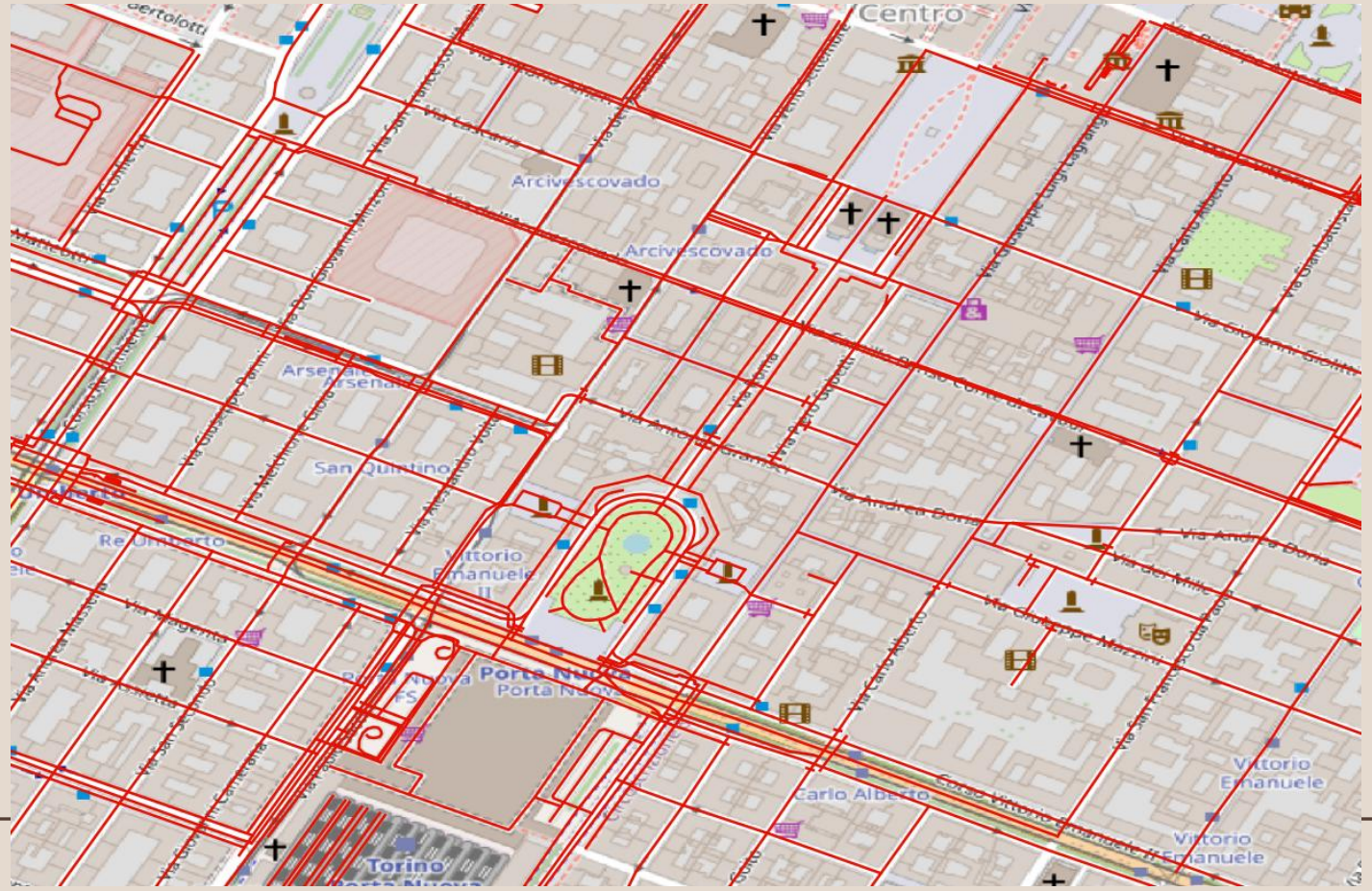
	osm_id	opening_ho	name	shop
1	4736057602	Mo-Sa 09:00-19...	LillaPois	convenience
2	9660117717	Mo-Su 07:30-20...	Carrefour Express	convenience
3	10951523952	Mo-Su 10:00-23...	Bhuiyan Mini ...	convenience
4	5113659500	*NULL*	Silvia Narese ca...	shoes
5	7236283370	PH,Mo-Su 08:0...	Legami Milano	stationery
6	11104708566	Mo-Fr 08:00-20:...	Yves Rocher	beauty
7	2018127864	*NULL*	Sephora	cosmetics
8	2018127868	*NULL*	nicla	clothes
9	4062467288	Mo-Sa 08:00-22...	Carrefour Express	convenience
10	9785576834	*NULL*	MyEbike	bicycle

Each point is associated with a specific record in the attribute table

Some examples

Polyline geometry

Red lines represent streets centerline geometry



	osm_id	surface	name	oneway	highway
1	8272429	asphalt	Corso Re Umberto	NULL	secondary
2	22907794	paving_stones	Via Roma	no	residential
3	22908219	paving_stones	Piazza Carlo Felice	yes	residential
4	22908320	asphalt	Via Venti Settembre	yes	service
5	22908426	sett	Via Carlo Alberto	NULL	pedestrian
6	23015864	asphalt	Via Goito	yes	residential
7	23015870	asphalt	Via Sant'Anselmo	yes	residential
8	23016548	NULL	NULL	yes	service
9	23016551	NULL	Piazza Giuseppe Luigi Lagrange	yes	service
10	23016554	cobblestone:fla...	Via Guarino Guarini	yes	residential
11	23016555	asphalt	Via Urbano Rattazzi	yes	residential
12	23016557	asphalt	Piazza Giuseppe Luigi Lagrange	yes	service
13	23026369	asphalt	Via Camillo Benso Conte di Cavour	yes	residential

Each line is associated with a specific record in the attribute table

Some examples

Polygon geometry

Blue areas represent buildings geometry

	osm_id	building	name	opening_ho	shop
1	315435507	commercial	Vincenzi Servizi ...	Mo-Fr 08:30-12:...	NULL
2	22983909	train_station	Torino Porta Nu...	NULL	NULL
3	250151807	school	Scuola Media S...	NULL	NULL
4	247632629	church	San Secondo M...	NULL	NULL
5	191379103	yes	San Filippo Neri	NULL	NULL
6	188466381	yes	Reale Società Gi...	NULL	NULL
7	191213695	yes	Principi di Piem...	NULL	NULL
8	264603315	yes	Palazzo Valperg...	NULL	NULL
9	3555630	yes	Palazzo Lascaris	NULL	NULL
10	23054067	yes	Palazzo dell'Ac...	NULL	NULL
11	3988223	yes	Palazzo dell' Ar...	NULL	NULL
12	256087617	office	Palazzo degli Af...	NULL	NULL

Each polygon is associated with a specific record in the attribute table

GIS DATA FORMAT

GIS data formats

Data formats determine how data is **structured** and **stored**.

There are multiple GIS data formats for:

- Vector data
- Raster data

Vector data formats

Most common formats:

- shapefile (standard de facto)
- dxf/dwg (CAD)
- kml/kmz (Google Earth)
- gpx (a GPS Exchange Format)
- GeoJSON
- osm (OpenStreetMap)
- ...
- GeoDatabase
- GeoPackage

GIS vector data formats

- **Early GIS** systems separated spatial data from attribute data, and linked related records through a **common identifier**:
  - Generic file structures.
  - Specialized file structures (i.e. ESRI Shapefile).
- In **late 80's**, first developments for storing geographic data in modified Relational DataBase Management Systems (**RDBMS**).
- In **late 90's**, introduction of **geometry types and functions** in RDBMS (Oracle Spatial, PostGIS, etc.).

Generic file structures

Text Files: by including a numeric fields for X and Y coordinates, you can associate **references to point entities in a text file**.

Note that the way that these coordinates relate to places on the Earth requires some **metadata** that specifies the specific Coordinate Reference System that is employed.

```
name,lat,lon,alt,date,real_val,norm_val
ALBA TANARO,44.70528,8.0275,172.0,2003-02,7.40357142857143,-0.19763636638045
ALBA TANARO,44.70528,8.0275,172.0,2003-03,16.3677419354839,0.965331811240329
ALBA TANARO,44.70528,8.0275,172.0,2003-04,17.7033333333333,1.13860498724754
ALBA TANARO,44.70528,8.0275,172.0,2003-05,25.2032258064516,2.11160480637061
ALBA TANARO,44.70528,8.0275,172.0,2003-06,32.2433333333333,3.0249543476894
ALBA TANARO,44.70528,8.0275,172.0,2003-07,31.9161290322581,2.98250444232028
ALBA TANARO,44.70528,8.0275,172.0,2003-08,33.9774193548387,3.24992629112942
ALBA TANARO,44.70528,8.0275,172.0,2003-09,23.8766666666667,1.93950343189183
ALBA TANARO,44.70528,8.0275,172.0,2003-10,15.4838709677419,0.850662661672705
ALBA TANARO,44.70528,8.0275,172.0,2003-11,10.83,0.246891924150086
ALBA TANARO,44.70528,8.0275,172.0,2003-12,7.61666666666667,-0.169990419606436
```

Specialized file structures - shapefile

A shapefile is a digital **vector** storage format for storing **geometric location** and associated **attribute information.**

This file format is **proprietary** (developed and regulated by Esri) but **open** and its technical specifications are published and can be implemented and used freely.

Because of its wide use, shapefile has become a **de facto standard** for creation and interchange of vector geospatial data.

Shapefile format

Points, lines and polygons to extend the range of datatypes.

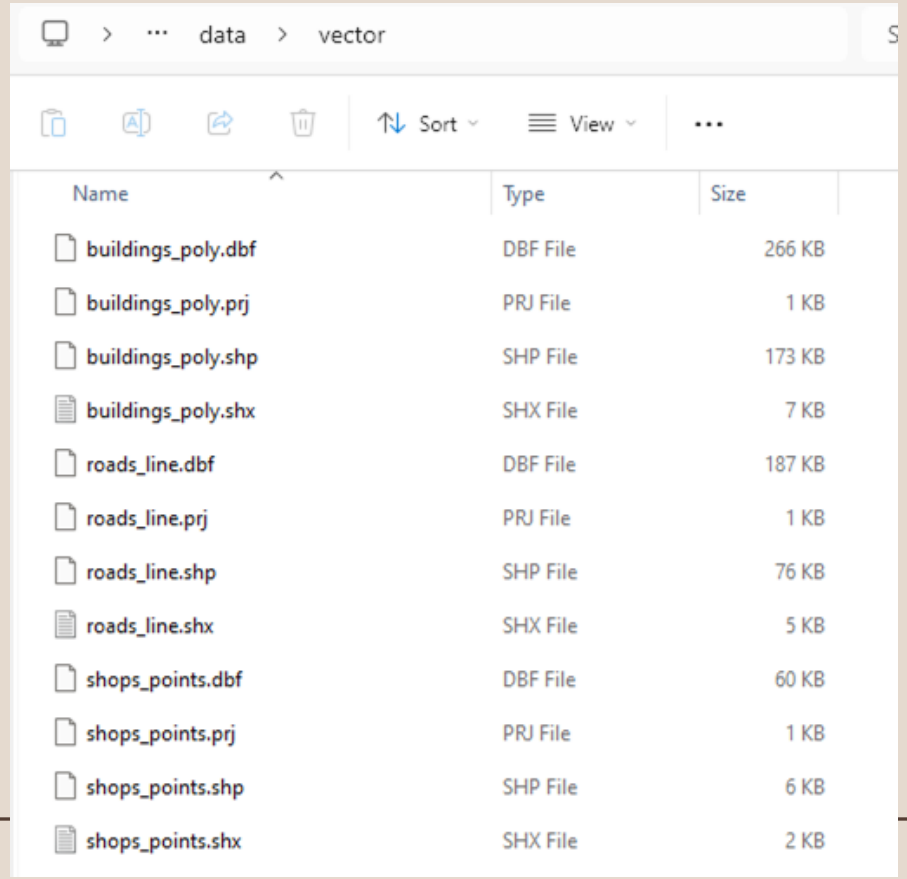
A **multifile** format (files located in the same folder):

Mandatory (see Shapefile Overview)	Optional
• .shp — **shape format**: the feature geometry itself. • .shx — **shape index** format: a positional index of the feature geometry to allow seeking forwards and backwards quickly. • .dbf — **attribute format**: columnar attributes for each shape, in dBase IV format.	• .prj — **projection format:** the coordinate system and projection information, a plain text file describing the projection using well-known text format. • .sbn and .sbx — a **spatial index** of the features. • .shp.xml — **geospatial metadata** in XML format, such as ISO 19115 or other XML schema. • ...

Shapefile format

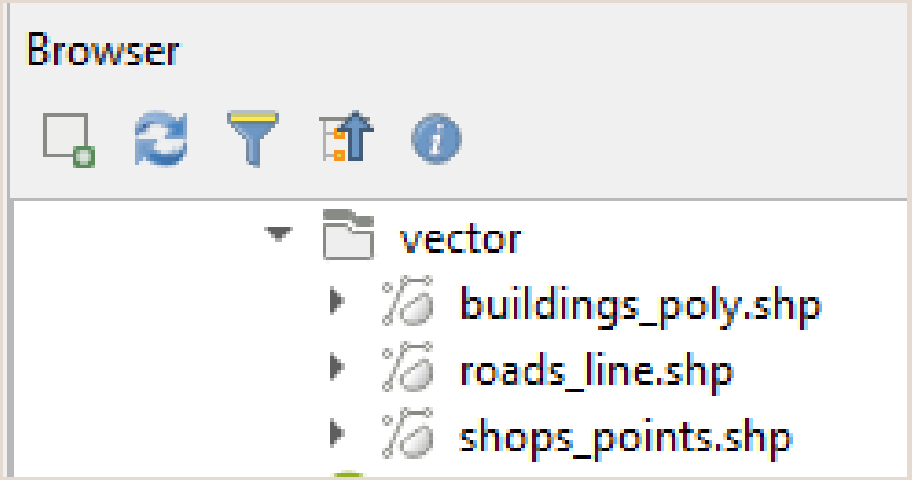
Shapefile view on

Windows file explorer



Shapefile view on

QGIS file browser



Shapefile format

PROS

- Widely supported format.
- Proprietary, but specification is open.
- For many use cases, it is good enough:
  - Good reading performance, thanks to the geometry indexing file (shx).
  - Relatively efficient in terms of file size: smaller compared to text-based format.

The number of items in the lists suggests staying away from shapefiles but some of the listed disadvantages are not relevant in most use cases.
NEVERTHELESS
why work with a limited format if there are alternatives available?

CONS

- No coordinate reference system definition (.prj is optional).
- Multifile format (error prone).
- Attribute names limited to 10 characters.
- Only 255 attributes (DBF limitation).
- Limited fields data types: float, integer, date and text (max 254 char.).
- Unknown character set (UTF8? WIN1252?).
- It's limited to 2-4GB of file size.
- No topology (no complex geometry relationships).
- Single geometry type per file.
- "Flat table" format (no hierarchies, relations or tree structure).
- 3D data with textures not supported.
- Projections definition inconsistencies: Esri WKT definitions are sometimes incompatible (or lack parameters) compared to standard definitions in EPSG.
- No NULL value: zero and no data cannot be distinguished for numerical fields.

Source: Shapefile must die! (switchfromshapefile.org)

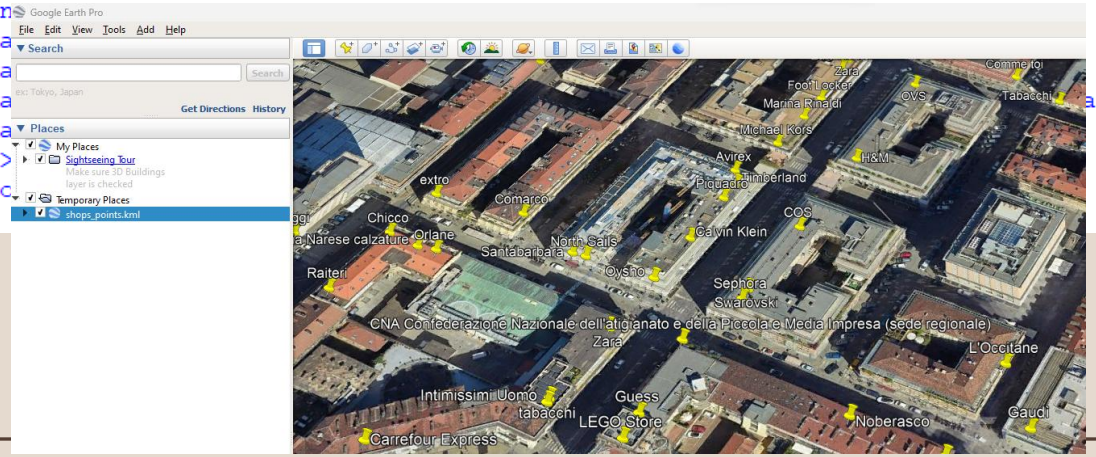
Specialised file structures – KML/KMZ

KML stands for Keyhole Markup Language. This GIS format is XML-based and is primarily used for Google Earth.

KMZ (KML-Zipped) replaced KML as being the default Google Earth geospatial format, being a **compressed version** of the file.

KML/KMZ became an **international standard** of the Open Geospatial Consortium in 2008.

```
<?xml version="1.0" encoding="utf-8" ?>
<kml xmlns="http://www.opengis.net/kml/2.2">
<Document id="root_doc">
<Schema name="planet_osm_point" id="planet_osm_point">
    <SimpleField name="osm_id" type="string"></SimpleField>
    <SimpleField name="opening_hours" type="string"></SimpleField>
    <SimpleField name="shop" type="string"></SimpleField>
    <SimpleField name="building" type="string"></SimpleField>
</Schema>
<Folder><name>planet_osm_point</name>
  <Placemark>
    <name>Comarco</name>
    <ExtendedData><SchemaData schemaUrl="#planet_osm_point">
        <SimpleData name="osm_id">1167321007</SimpleData>
        <SimpleData name="shop">estate_agent</SimpleData>
    </SchemaData></ExtendedData>
      <Point><coordinates>7.6797346,45.065164</coordinates></Point>
  </Placemark>
  <Placemark>
    <name>Lidl</n
    <ExtendedData
        <SimpleDa
        <SimpleDa
        <SimpleDa
    </SchemaData>
      <Point><coo
  </Placemark>
```

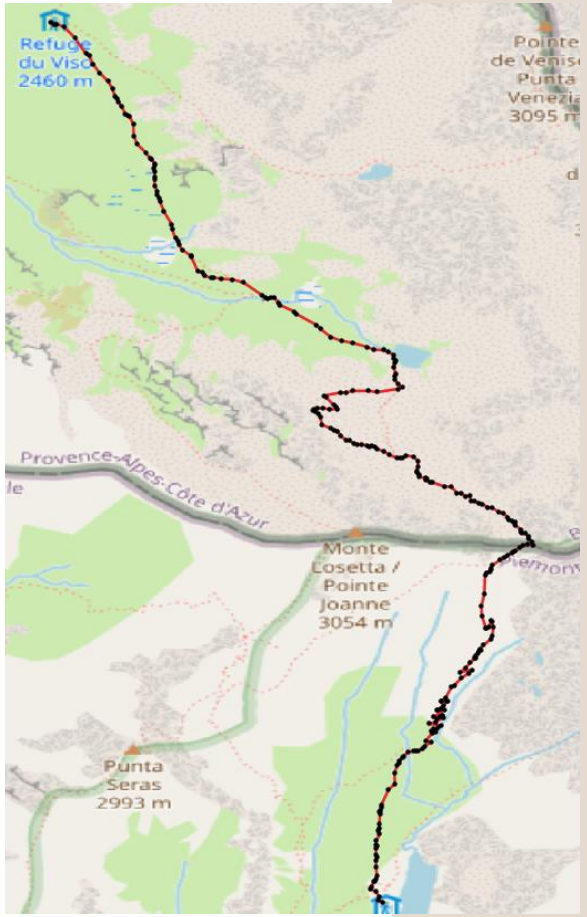


Specialised file structures – GPX

GPX - GPS Exchange format is an XML schema that describes waypoints, tracks, and routes captured from a GPS receiver.

The minimum requirement are **latitude and longitude** coordinates, optionally location properties including **time, elevation** and geoid height as tags.

```
<?xml version='1.0' encoding='UTF-8'?>
<gpx version="1.1" creator="https://www.komoot.de" xmlns="http://www.topografix.com
  <metadata>
    <name>Dal Refuge du Viso al Rifugio Vallanta</name>
    <author>
      <link href="https://www.komoot.de">
        <text>komoot</text>
        <type>text/html</type>
      </link>
    </author>
  </metadata>
  <trk>
    <name>Dal Refuge du Viso al Rifugio Vallanta</name>
    <trkseg>
      <trkpt lat="44.700346" lon="7.051230">
        <ele>2451.789225</ele>
        <time>2023-07-18T09:27:18.045Z</time>
      </trkpt>
      <trkpt lat="44.700269" lon="7.051356">
        <ele>2451.789225</ele>
        <time>2023-07-18T09:27:33.147Z</time>
      </trkpt>
      <trkpt lat="44.700297" lon="7.051442">
        <ele>2451.789225</ele>
        <time>2023-07-18T09:27:41.843Z</time>
      </trkpt>
      <trkpt lat="44.700195" lon="7.051748">
        <ele>2451.789225</ele>
        <time>2023-07-18T09:28:13.043Z</time>
      </trkpt>
      <trkpt lat="44.699751" lon="7.052183">
        <ele>2451.400627</ele>
        <time>2023-07-18T09:29:25.387Z</time>
      </trkpt>
      <trkpt lat="44.699121" lon="7.052612">
        <ele>2447.359917</ele>
        <time>2023-07-18T09:31:01.949Z</time>
      </trkpt>
```



Specialised file structures – GeoJSON

Mostly for **web-based mapping**.

GeoJSON stores coordinate as text in JavaScript Object Notation (JSON) form.

GeoJSON has a straightforward syntax that you can modify in any **text editor**.

Webmaps browsers understand JavaScript so by default GeoJSON is a common web format.


```
{
  "type": "FeatureCollection",
  "name": "shops_points",
  "crs": {
    "type": "name",
    "properties": {
      "name": "urn:ogc:def:crs:OGC:1.3:CRS84"
    }
  },
  "features": [
    {
      "type": "Feature",
      "properties": {
        "osm_id": 4736057602,
        "opening_hours": "Mo-Sa 09:00-19:30; Su 10:00-13:00,15:30-19:30",
        "name": "LillaPois",
        "shop": "convenience"
      },
      "geometry": {
        "type": "Point",
        "coordinates": [
          7.6842789,
          45.0656023
        ]
      }
    },
    {
      "type": "Feature",
      "properties": {
        "osm_id": 9660117717,
        "opening_hours": "Mo-Su 07:30-20
        "name": "Carrefour Express",
        "shop": "convenience"
      },
```


Specialised file structures - osm (OpenStreetMap)

OSM files are the native file for **OpenStreetMap**. OSM is an **XML-based** file format.

QGIS can load native OSM files.


```
<node id="1168929257" visible="true" version="2" changeset="128364593" timestamp="2022-11-01T21:36:53Z" user="EdoBoo" uid="9522091" lat="45.0652615" lon="7.6792422">
 <tag k="crossing" v="traffic_signals"/>
 <tag k="highway" v="crossing"/>
</node>
```

The more efficient, smaller PBF Format ("Protocolbuffer Binary Format") is an alternative to the XML-based format.

Other vector data formats - Spatial Database

A **database** is a collection of **connected information** that allows entry, storage, input, output, and **organization of data**.

Spatial databases (aka geodatabases, geospatial databases) provide a strong foundation for accessing, storing, and managing **spatial (and geographic) data**.

The **database management system** (DBMS) is the software component that serves as an **interface** between users and their databases.

Other vector data formats - Spatial Database

Specifically, spatial databases are built on top of **relational database management systems (RDBMS)**, which reduce redundancy (repetitive information that wastes storage space) through **normalisation** (linking tables together through primary and foreign keys).

It is important to make the distinction between **typical databases** and **spatial databases**.

A **spatial database** stores **feature geometry** that describes **shape** and **location**. The geometry of spatial features is compressed and stored in a binary field along with the **attribute** data that describe the feature.

Other vector data formats - Spatial Database

The primary advantage of spatial databases, over file-based data storage, is that they let a GIS build on the existing **capabilities of RDBMS**:

- **support for SQL** and the ability to generate complex geospatial queries.
- database's client/server architecture which supports **multiple users** simultaneously and lets them view, edit, and query the database without conflict.

Other vector data formats - Spatial Database

Pros:

- RDBMS **scalability** (manage big data volumes, several users and different roles, etc.)
- RDBMS **performances.**
- Increased **functionalities** (geometric networks, topology, cartographic representations, dimensions, and raster data management).
- Data **integrity** (i.e. domains, subtypes, relationships, constraints).
- Solid framework for data **management** (i.e. multiediting, replica and synchronization, historical archives).

Cons:

- additional **complexity** (and need for specific competences) for installing and maintaining the DBMS component.

Other vector data formats - ESRI GeoDatabase

ESRI File GeoDatabase (.gdb)

Esri created the file geodatabase (FGDB) to be a **container** for storing **multiple attribute tables, vector and raster data sets**. They have fast performance, versatile relationships, compatible storage for rasters, improved spatial indexes, data compression, customizable configuration, and 1 terabyte file size restrictions. Geodatabases can **store more complex data** such as networks, raster mosaics, and feature data sets.

QGIS users can **read** file geodatabases by dragging the geodatabases, with extension .gdb, directly into the Layers pane, thanks to the open source read-only GDAL driver for the file geodatabase format: OpenFileGDB. FGDB rasters and complex data are not yet supported.

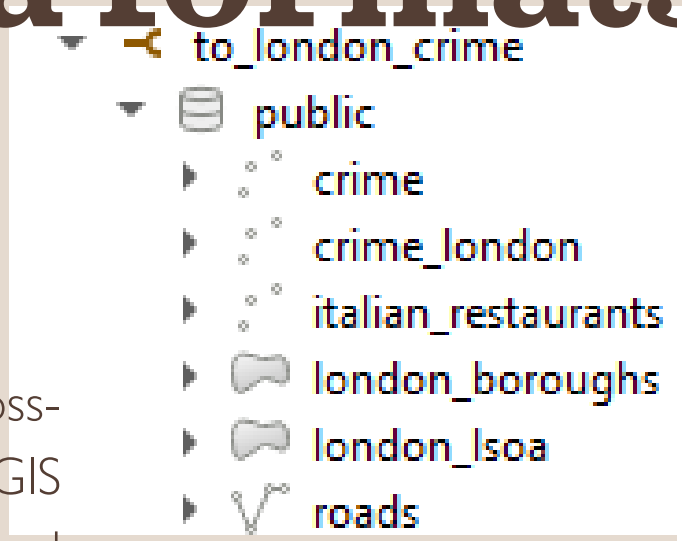
Other vector data formats - GeoDatabase

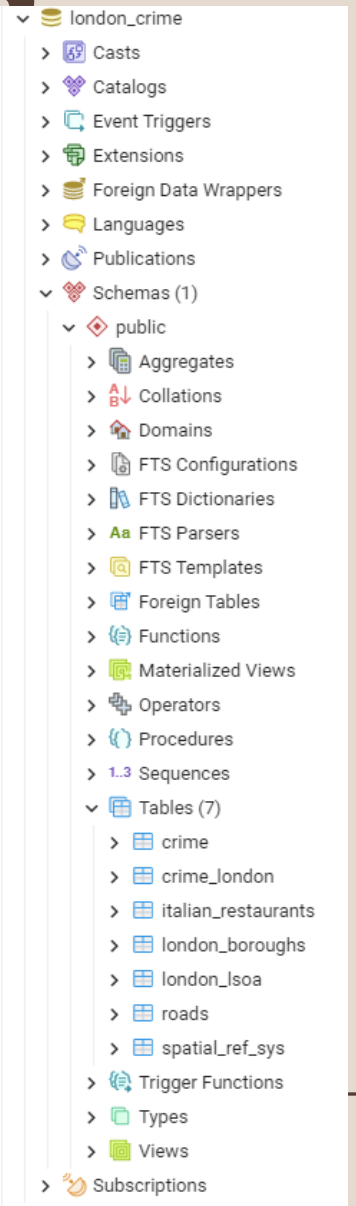
PostgreSQL + PostGIS

Open source **PostGIS adds spatial capabilities** to the cross-platform **PostgreSQL database**. The three features that PostGIS delivers to PostgreSQL DBMS are spatial types, indexes and functions.

With support for different geometry types, the PostGIS spatial database allows **querying and managing** information about locations and mapping.

PostGIS can be leveraged in several GIS software packages including **QGIS**.





Other vector data formats - GeoPackage

GeoPackage is an open, standards-based, platform-independent, portable, self-describing, compact format for transferring geospatial information.

The **GeoPackage Encoding Standard** describes a set of **conventions** for storing within an **SQLite** database the following entities:

- vector features.
- tile matrix sets of imagery and raster maps at various scales.
- attributes (non-spatial data).
- extensions.

A **GeoPackage is** the SQLite **container** and the **GeoPackage Encoding Standard governs the rules** and requirements of content stored in a GeoPackage container. **The GeoPackage standard defines the schema for a GeoPackage**, including table definitions, integrity assertions, format limitations, and content constraints.

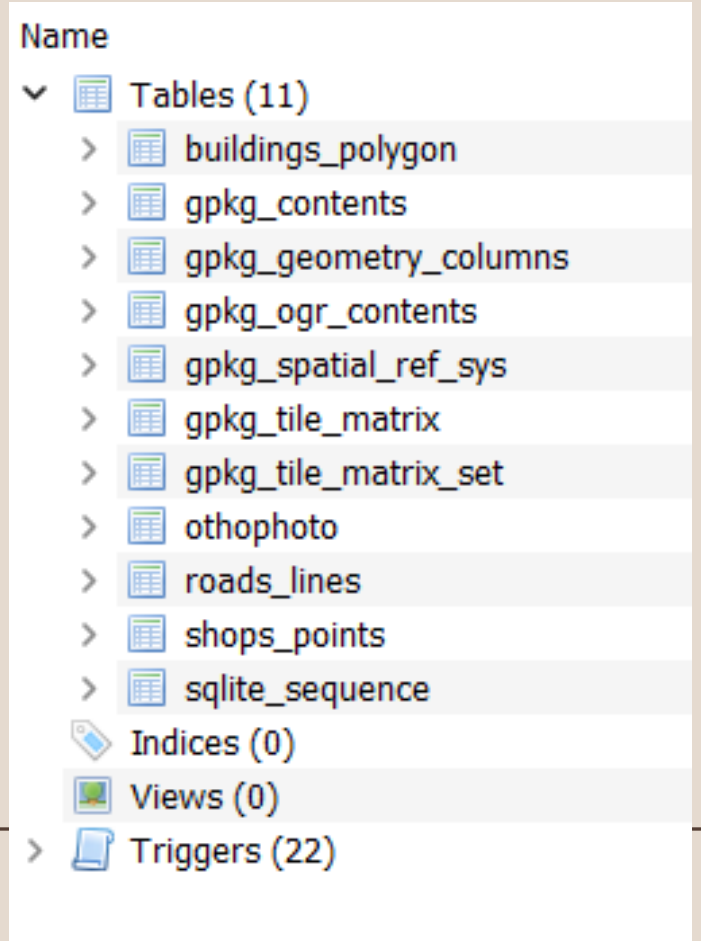
Other vector data formats - GeoPackage

A GeoPackage is composed by:

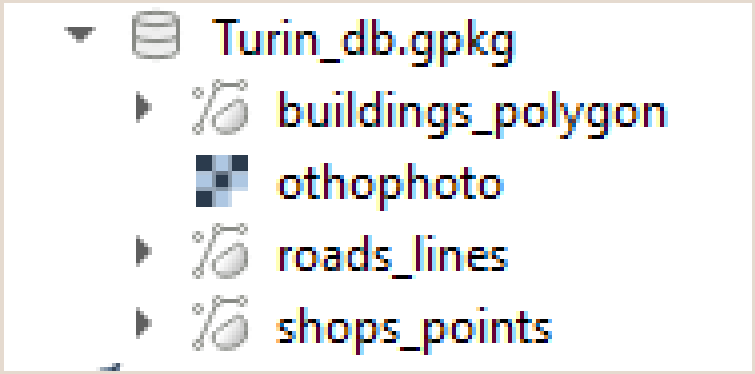
- An SQLite **container**
- GeoPackage **Encoding Standard** that governs the rules and requirements of content
  - GeoPackage **schema** (table definitions, integrity assertions, format limitations, content constraints
  - **extension** mechanism provides a way to include additional functionality in GeoPackages
- Several GeoPackage implementations are available
  - DB Browser for SQLite - direct SQL interface
  - GDAL, QGIS and ESRI - data manipulation and GIS

GeoPackage

Geopackage view in SQLite



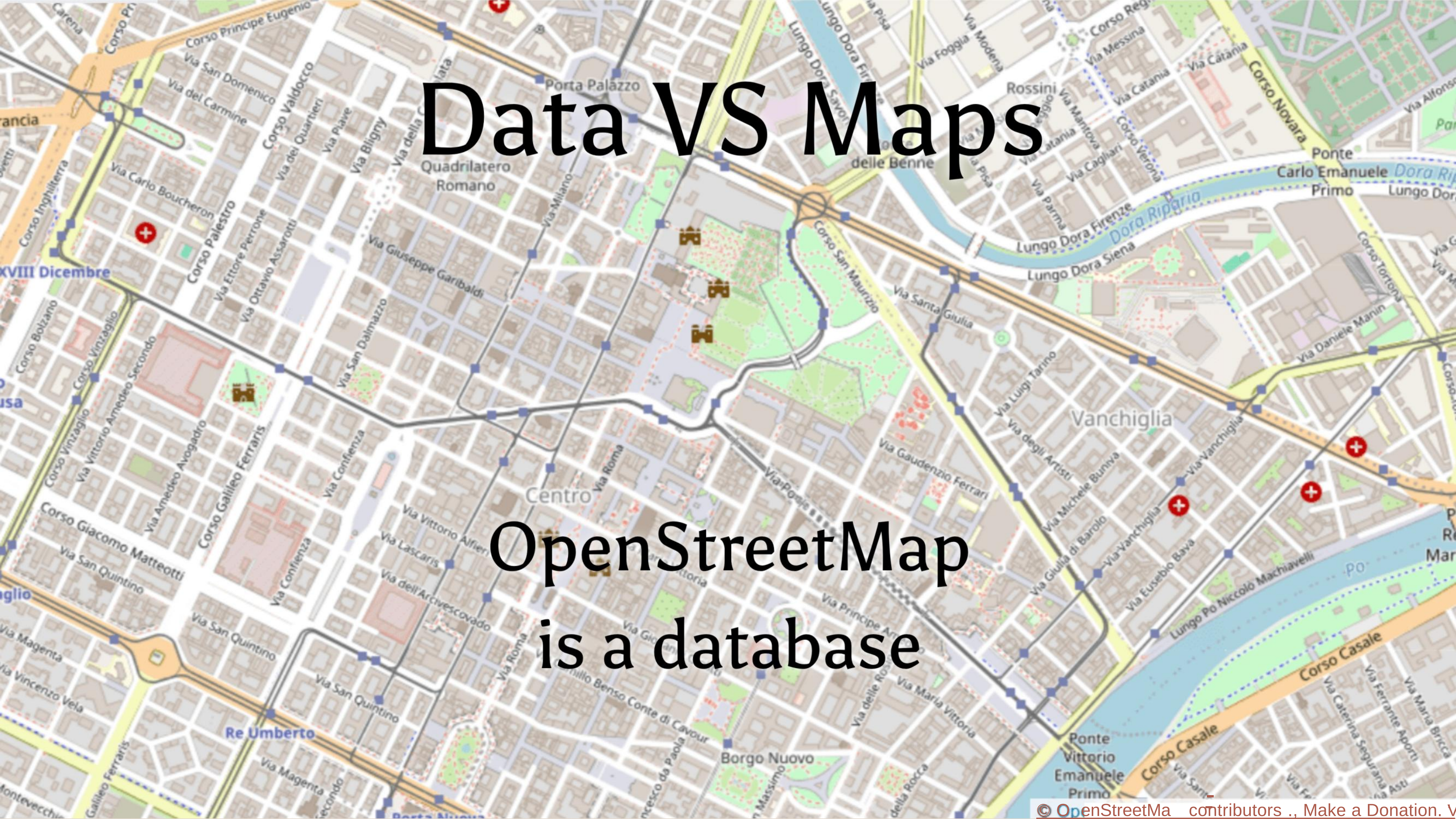
Geopackage view in QGIS file browser



OPENSTREETMAP


OpenStreetMap map
• Database of geographical data
• open and free
• The Wikipedia of maps
© OpenStreetMap contributors ♥ Make a Donation.

Data VS Maps



OpenStreetMap is a database

A bit of history...

Project designed by **Steve Coast** in **2004**

Spatial data created and controlled
by **private** actors and **governmental bodies**

Expensive for their creators and their users
Restrictive on redistribution terms

Difficult to keep up-to-date

A bit of history...

Geographical data **created** and **maintained**
by the **community**
in a ***voluntary and collaborative*** approach
(Wikipedia model)

Distributed under a **free license**
Updated in real time

Everyone can contribute!

Why OpenStreetMap?

Data created by

public institutions:

- rare updates
- sometimes with fees

Why OpenStreetMap?

Data created by

private companies:

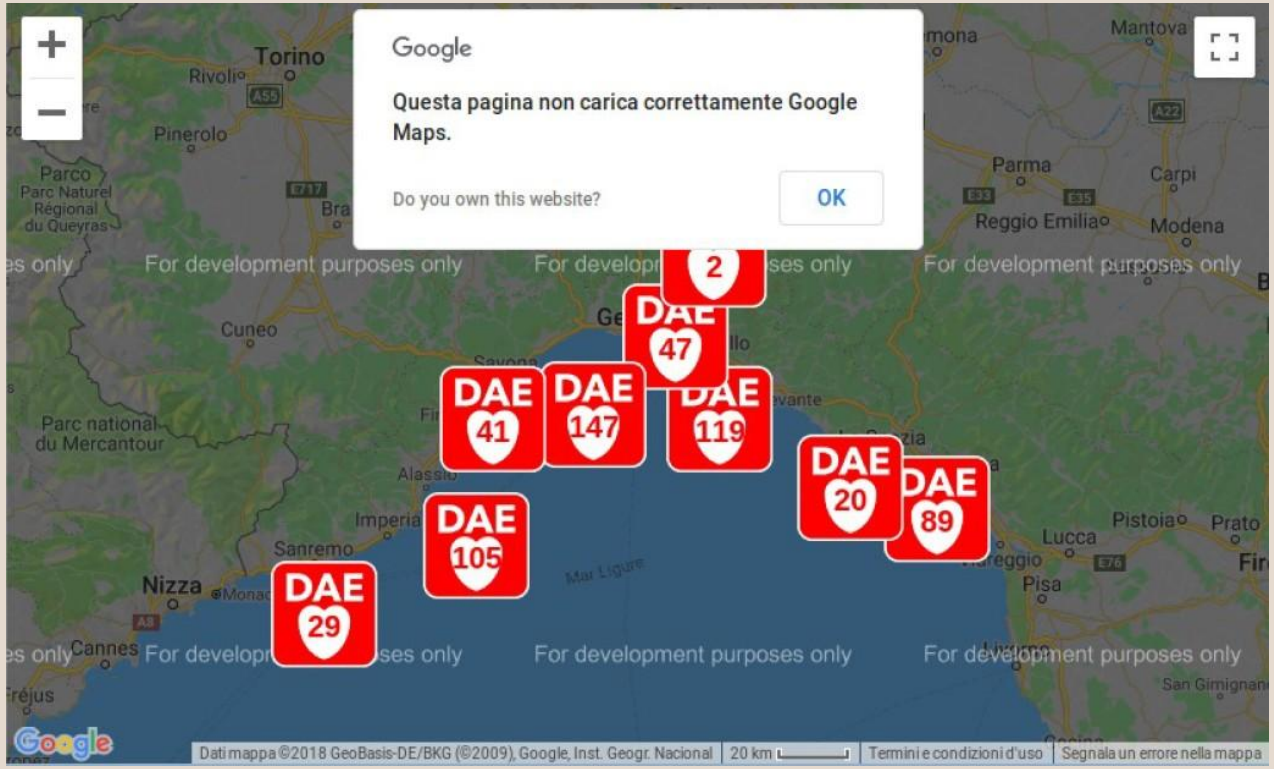
- regular updates
- high costs

Why OpenStreetMap?

Data created by

private companies:

- restricted usages

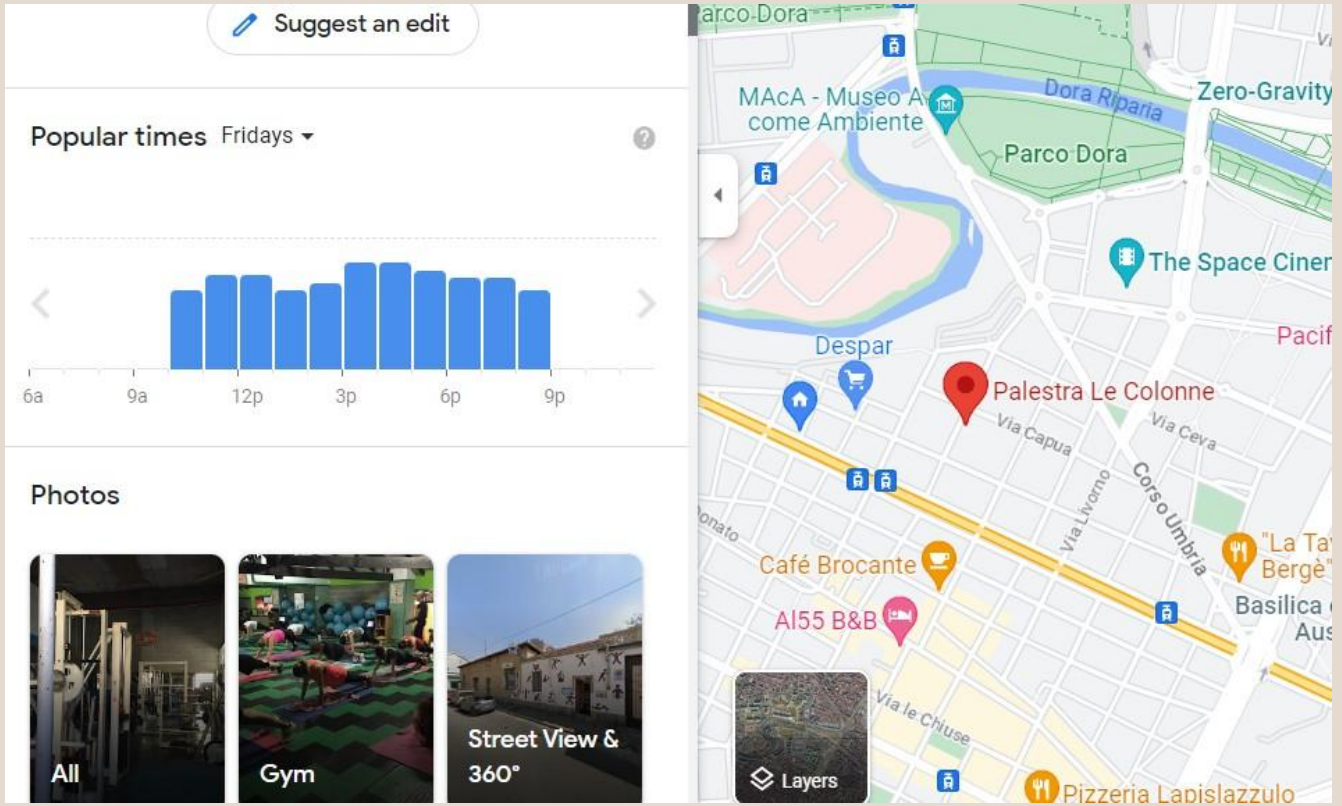


Why OpenStreetMap?

Data created by

private companies:

- restricted usages



Why OpenStreetMap?

By René Magritte(1898-1967) - Image taken from a University of Alabama site, "Approaches to Modernism": [1], Fair use (Old-50), https://en.wikipedia.org/w/index.php?curid=555365

Maps,
and the geographic data
on which they are based,
are always only a
representation
of the real world

Why OpenStreetMap?

Relying on a single (and private)

map source

means being dependent
on what the map owner
decides to display or not

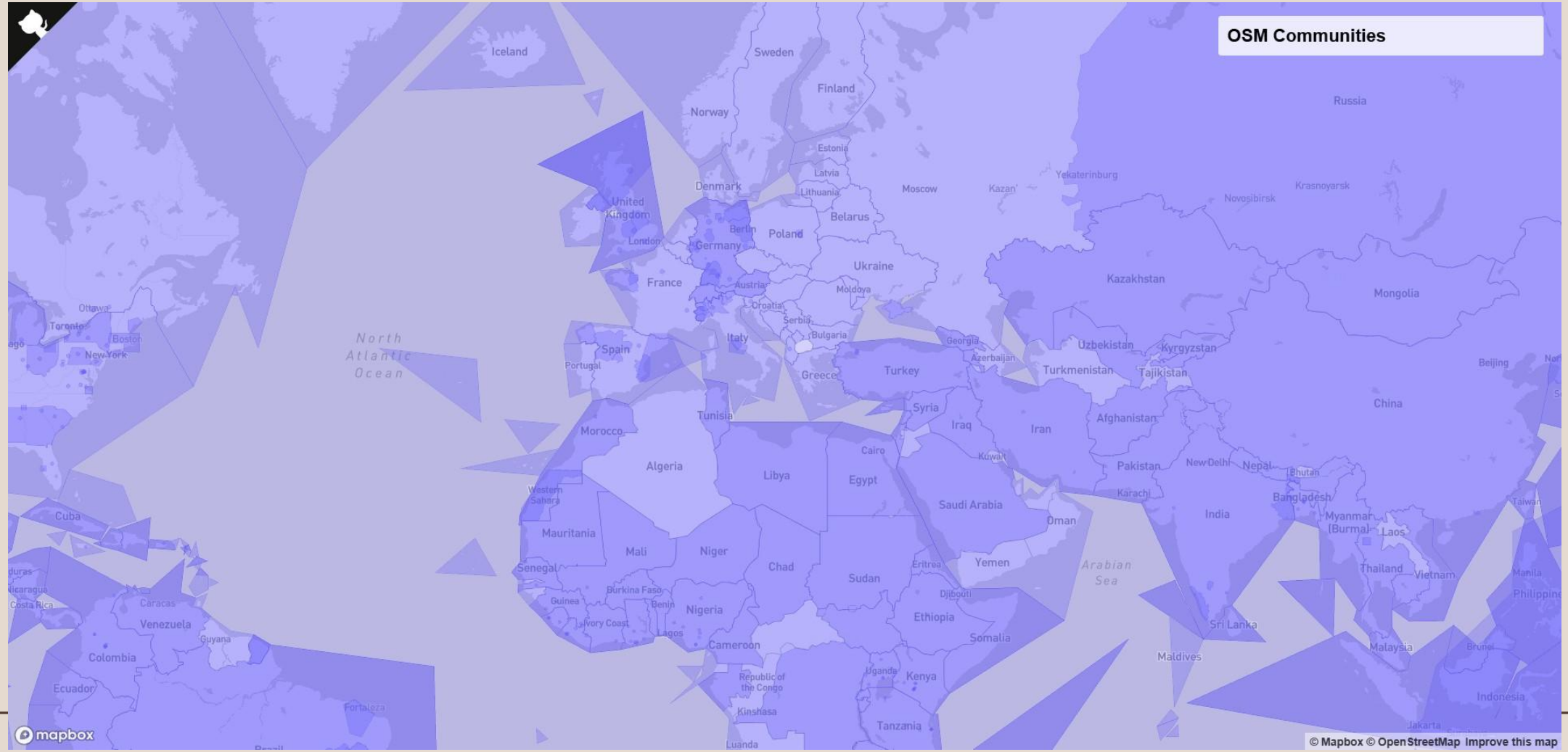
Why OpenStreetMap?

A collaborative map allows
for a "debated/concerted" view
that is the result of different voices

Mapping is a community activity!

The community

The community



Who uses OpenStreetMap

Universities and Research Institutions

World Bank, UN, MSF, Red Cross

Many national, regional and municipal institutions

National Fire Brigade

Civil Protection Department

ISTAT, Revenue Agency, Carabinieri

Who uses OpenStreetMap

facebook

amazon

mapbox

The Washington Post

TELENAV

Bing

lonely planet

+ a b | e a u

TESLA

Grab

The New York Times

STRAVA

Who uses OpenStreetMap

What can be mapped

Transport infrastructures

Service structures

Points of interest

Natural elements

Land uses

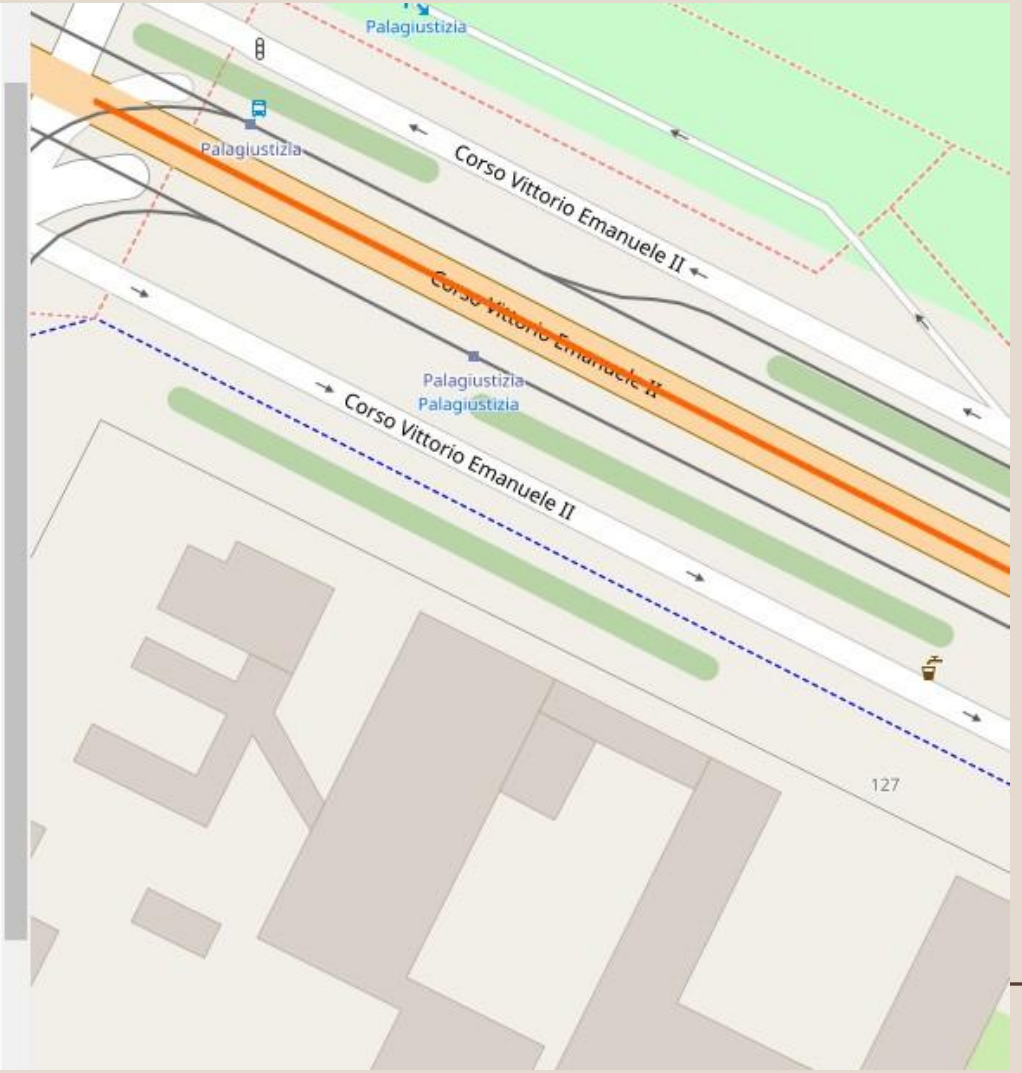
...

What can be mapped

Adding details and specifying characteristics

Etichette

cycleway	no
foot	no
highway	primary
lanes	2
lit	yes
maxspeed	50
name	Corso Vittorio Emanuele II
name:etymology:wikidata	Q168691
parking:both	no
parking:condition:both	no_parking
shoulder	no
sidewalk	none
source	TCI, Guida d'Italia, 1914
surface	asphalt



What can be mapped

Verifiable data

Another mapper

should be able to go to the same place and collect the same data ("verify" the data entered)

What cannot be mapped

Private information

Historical elements no longer present

Non-stable or temporary elements

Data obtained from proprietary sources

Mapping styles

Armchair mapping or remote mapping

tracing information by
photointerpretation of the satellite imagery

On field survey

using a GPS to record positions
using smartphone apps (OsmAnd, MapSwipe, StreetComplete...)
using Field Papers to print a map of an area, draw on it and add notes, and load the paper back into OSM editors

OpenStreetMap data model

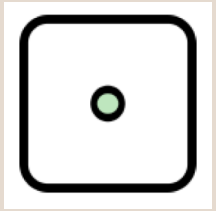
A **geographical entity**

is modelled by specifying

its geometry

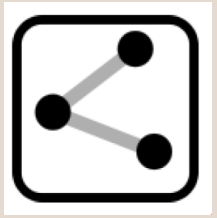
and its attributes

OSM Geometries



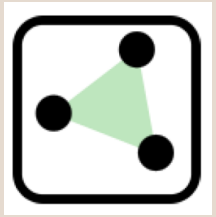
NODE

Defined by latitude, longitude and ID



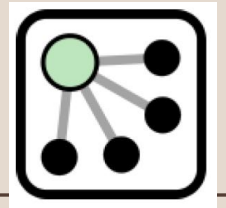
WAY

Ordered sequence of nodes



CLOSED WAY

Ordered sequence of nodes where the first and last node are the same



RELATION

Describe how different elements are related to compose complex elements

OSM Attributes

Any element must have

at least one tag

The **tag** consists of a

KEY=VALUE pair

Each element can have multiple

(potentially infinite) tags

Node example


```
▼<osm version="0.6" generator="CGImap 0.8.3 (2793080 spike-07.openstreetmap.org)"
 copyright="OpenStreetMap and contributors" attribution="http://www.openstreetmap.org/copyright"
 license="http://opendatacommons.org/licenses/odbl/1-0/">
  ▼<node id="639634750" visible="true" version="2" changeset="7634771" timestamp="2011-03-22T08:32:06Z"
   user="plush" uid="83472" lat="45.0656562" lon="7.6547484">
     <tag k="amenity" v="drinking_water"/>
   </node>
 </osm>
```

Way example

Percorso: Via San Paolo (107167209)

Versione #5

Add road surfaces

Modificato quasi 4 anni fa da frazzmark · Gruppo di modifiche #48447084

Etichette

highway	residential
name	Via San Paolo
surface	asphalt

Nodi

▼ 4 nodi

1232172723 (parte dei percorsi — Via San Paolo (552704373) e — Via San Paolo (43684741))

1239242543 (parte del percorso — Via Monginevro (552708120))

2929276954

1232172714 (parte del percorso — Via Vigone (38027665))


```
▼<osm version="0.6" generator="CGImap 0.8.3 (2448896
 spike-07.openstreetmap.org)" copyright="OpenStreetMap and
 contributors"
 attribution="http://www.openstreetmap.org/copyright"
 license="http://opendatacommons.org/licenses/odbl/1-0/">
  ▼<way id="107167209" visible="true" version="5"
   changeset="48447084" timestamp="2017-05-06T08:41:56Z"
   user="frazzmark" uid="1140415">
     <nd ref="1232172723"/>
     <nd ref="1239242543"/>
     <nd ref="2929276954"/>
     <nd ref="1232172714"/>
     <tag k="highway" v="residential"/>
     <tag k="name" v="Via San Paolo"/>
     <tag k="surface" v="asphalt"/>
   </way>
 </osm>
```

Closed way example

Percorso: 397540290

Versione #3

Add building types and move road to correct place

Modificato 6 mesi fa da that_dude · Gruppo di modifiche #89445642

Etichette

building	apartments
building:levels	4
height	14

Nodi

▶ 11 nodi


```
▼<osm version="0.6" generator="CGImap 0.8.3 (2650895 spike-
 07.openstreetmap.org)" copyright="OpenStreetMap and contributors"
 attribution="http://www.openstreetmap.org/copyright"
 license="http://opendatacommons.org/licenses/odbl/1-0/">
  ▼<way id="397540290" visible="true" version="3" changeset="89445642"
   timestamp="2020-08-15T13:15:57Z" user="that_dude" uid="11342958">
     <nd ref="4003289678"/>
     <nd ref="4003289679"/>
     <nd ref="4003289680"/>
     <nd ref="4003289681"/>
     <nd ref="4003289682"/>
     <nd ref="4003289683"/>
     <nd ref="4003289684"/>
     <nd ref="4003289685"/>
     <nd ref="4003289686"/>
     <nd ref="4003289687"/>
     <nd ref="4003289678"/>
     <tag k="building" v="apartments"/>
     <tag k="building:levels" v="4"/>
     <tag k="height" v="14"/>
   </way>
 </osm>
```

Relation example

Relazione: 56: Corso Tirreno (Grugliasco) - Largo Tabacchi (1706267)

Versione #112

Parking lanes

Modificato 3 mesi fa da Martj9 · Gruppo di modifiche #94311036

Etichette

direction	Largo Tabacchi
from	Grugliasco - Corso Tirreno
name	56: Corso Tirreno (Grugliasco) - Largo Tabacchi
network	Formula
opening_hours	Mo-Sa 04:50-00:44; PH 06:00-00:48
operator	Gruppo Torinese Trasporti
ref	56
route	bus
to	Largo Tabacchi
type	route

Membri

▸ 171 membri

Relazione: Cenisia (9168509)

Versione #6

not a boundary -> multipolygon

Modificato 12 mesi fa da Garmin-User · Gruppo di modifiche #81344995

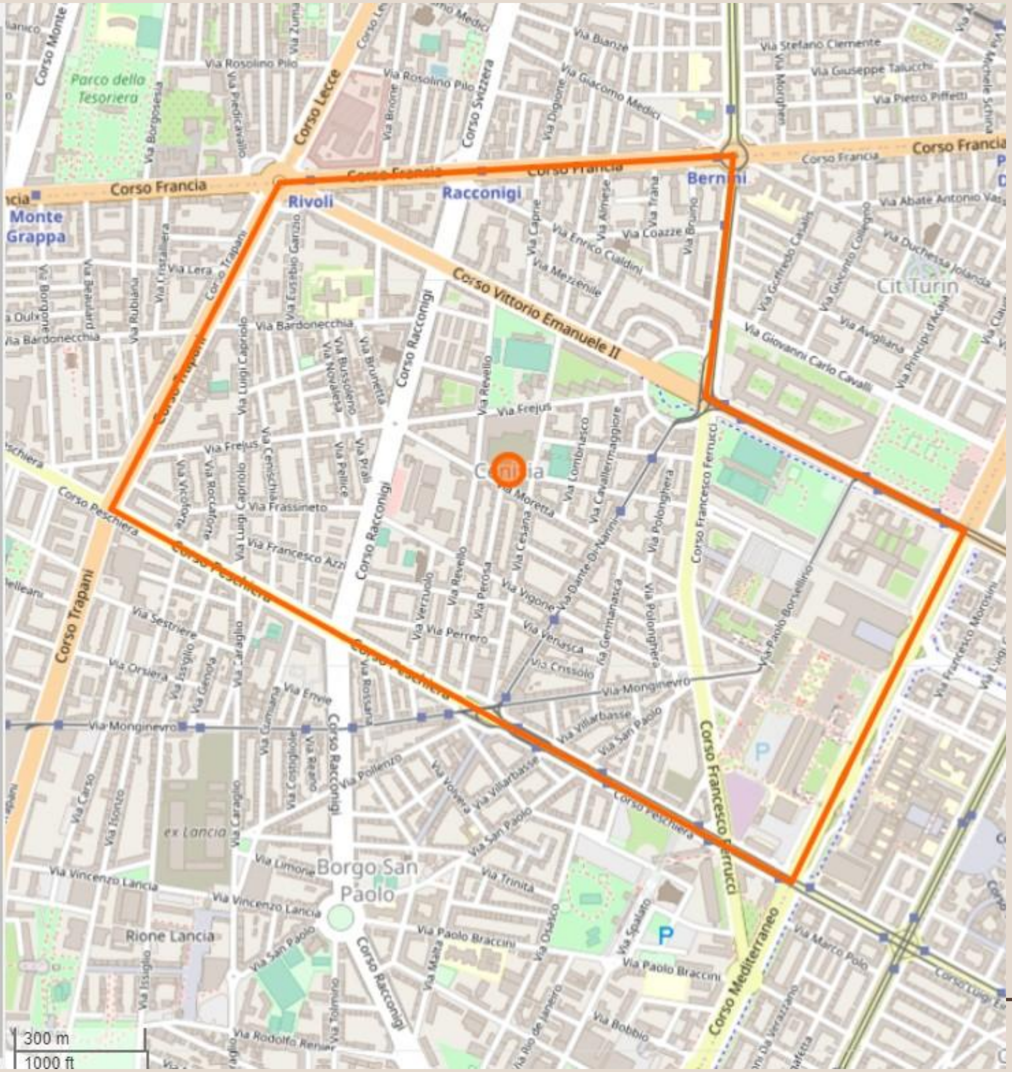
Etichette

description	Confini della parte occidentale e meridionale dell'ex Quartiere 5 della divisione a 23, dopo aver separato Cit Turin sui suoi confini storici (corsi Ferrucci e Vittorio).
name	Cenisia
place	suburb
type	multipolygon

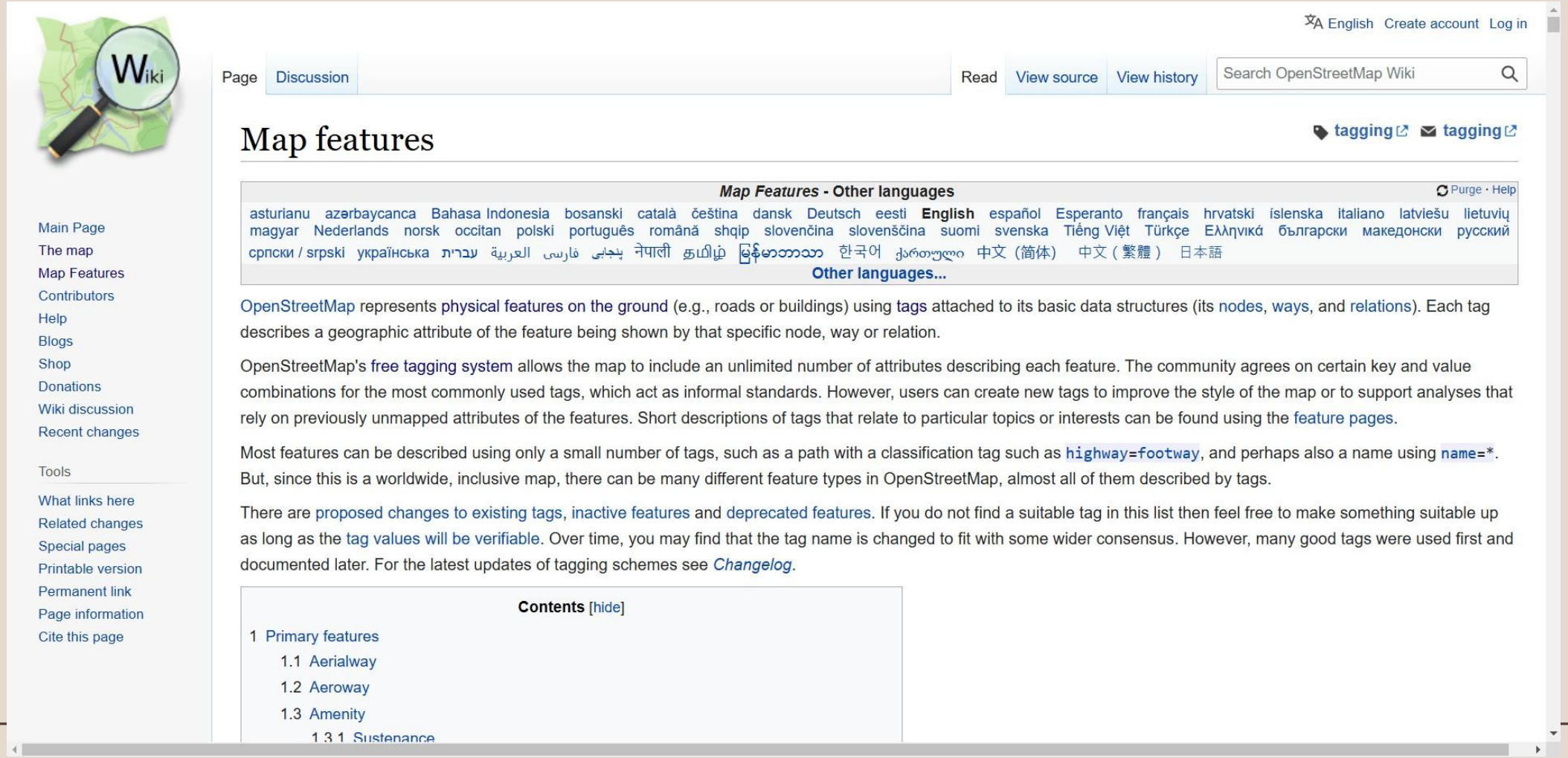
Membri

▸ 25 membri

Scarica XML · Visualizza cronologia



The answer is in the Wiki



English Create account Log in

Page Discussion | Read View source View history | Search OpenStreetMap Wiki

Map features

tagging tagging

***Map Features* - Other languages** Purge · Help

asturianu azərbaycanca Bahasa Indonesia bosanski català čeština dansk Deutsch eesti **English** español Esperanto français hrvatski íslenska italiano latviešu lietuvių magyar Nederlands norsk occitan polski português română shqip slovenčina slovenščina suomi svenska Tiếng Việt Türkçe Ελληνικά български македонски русский српски / srpski українська עברית العربية فارسی پنجابی नेपाली தமிழ் မြန်မာဘာသာ 한국어 ქართული 中文（简体） 中文（繁體） 日本語

Other languages...

Main Page
The map
Map Features
Contributors
Help
Blogs
Shop
Donations
Wiki discussion
Recent changes

Tools

What links here
Related changes
Special pages
Printable version
Permanent link
Page information
Cite this page

OpenStreetMap represents physical features on the ground (e.g., roads or buildings) using tags attached to its basic data structures (its nodes, ways, and relations). Each tag describes a geographic attribute of the feature being shown by that specific node, way or relation.

OpenStreetMap's free tagging system allows the map to include an unlimited number of attributes describing each feature. The community agrees on certain key and value combinations for the most commonly used tags, which act as informal standards. However, users can create new tags to improve the style of the map or to support analyses that rely on previously unmapped attributes of the features. Short descriptions of tags that relate to particular topics or interests can be found using the feature pages.

Most features can be described using only a small number of tags, such as a path with a classification tag such as `highway=footway`, and perhaps also a name using `name=*`. But, since this is a worldwide, inclusive map, there can be many different feature types in OpenStreetMap, almost all of them described by tags.

There are proposed changes to existing tags, inactive features and deprecated features. If you do not find a suitable tag in this list then feel free to make something suitable up as long as the tag values will be verifiable. Over time, you may find that the tag name is changed to fit with some wider consensus. However, many good tags were used first and documented later. For the latest updates of tagging schemes see *Changelog*.

Contents [hide]

1 Primary features
- 1.1 Aerialway
- 1.2 Aeroway
- 1.3 Amenity
  - 1.3.1 Sustenance

But also in TagInfo!

Why mapping indoor

Mapping the distribution of the interior spaces allows to:

- enable indoor navigation inside the Politecnico di Torino
- reuse the data for simulations on emergency planning and optimization of flow
- support the building maintenance and BIM(Building Information Model)

OSM indoor mapping

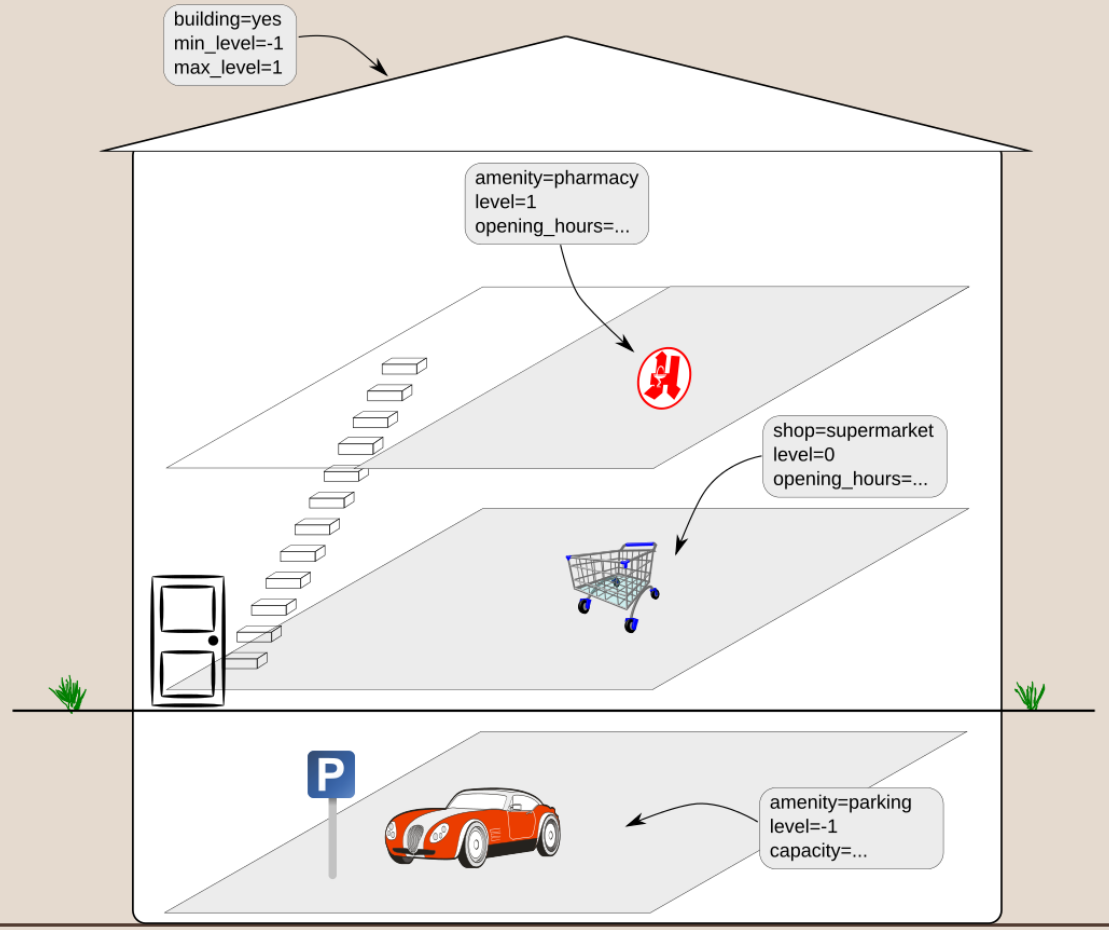
- OSM Simple Indoor Tagging schema

- It allows to model indoor spaces of **public buildings**:
- enabling indoor navigation/routing enabling floor plans and 3D maps visualization

OSM indoor mapping

Indoor tagging scheme
is designed to
be **compatible** with
the **3D building** taggingscheme

Indoor elements must be
inside the boundary
of a **3D building**



OSM indoor mapping

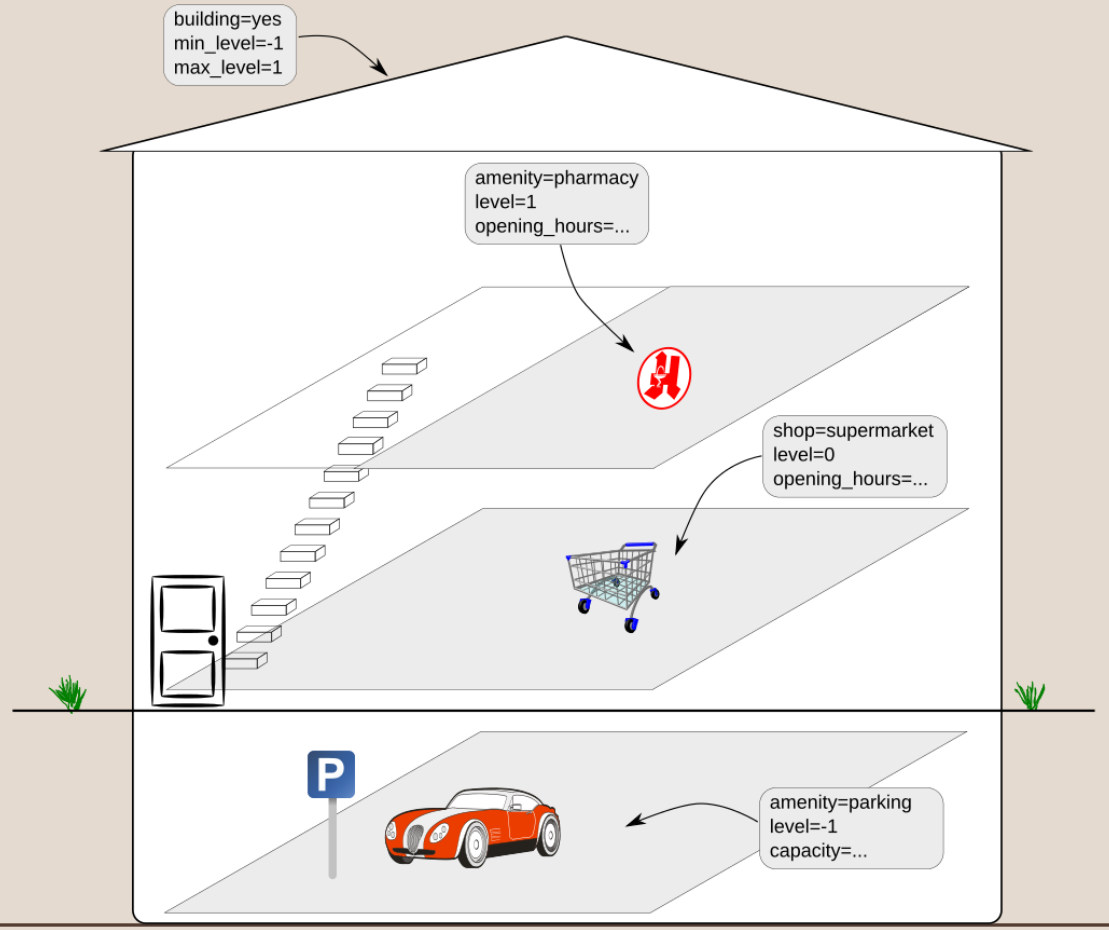
To define the

total number of building levels

the building feature

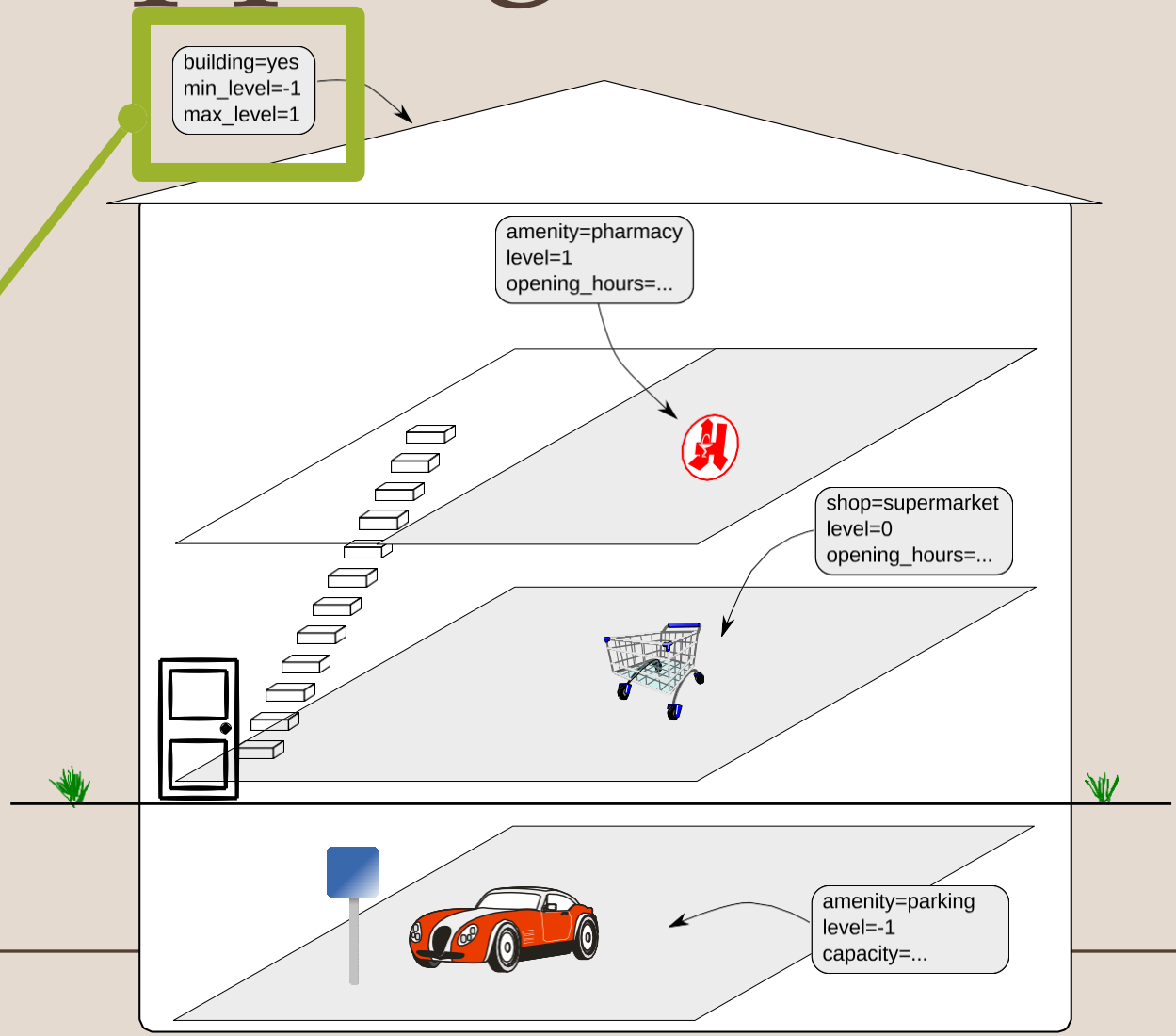
must have the tags

min_level=* and max_level=*



OSM indoor mapping

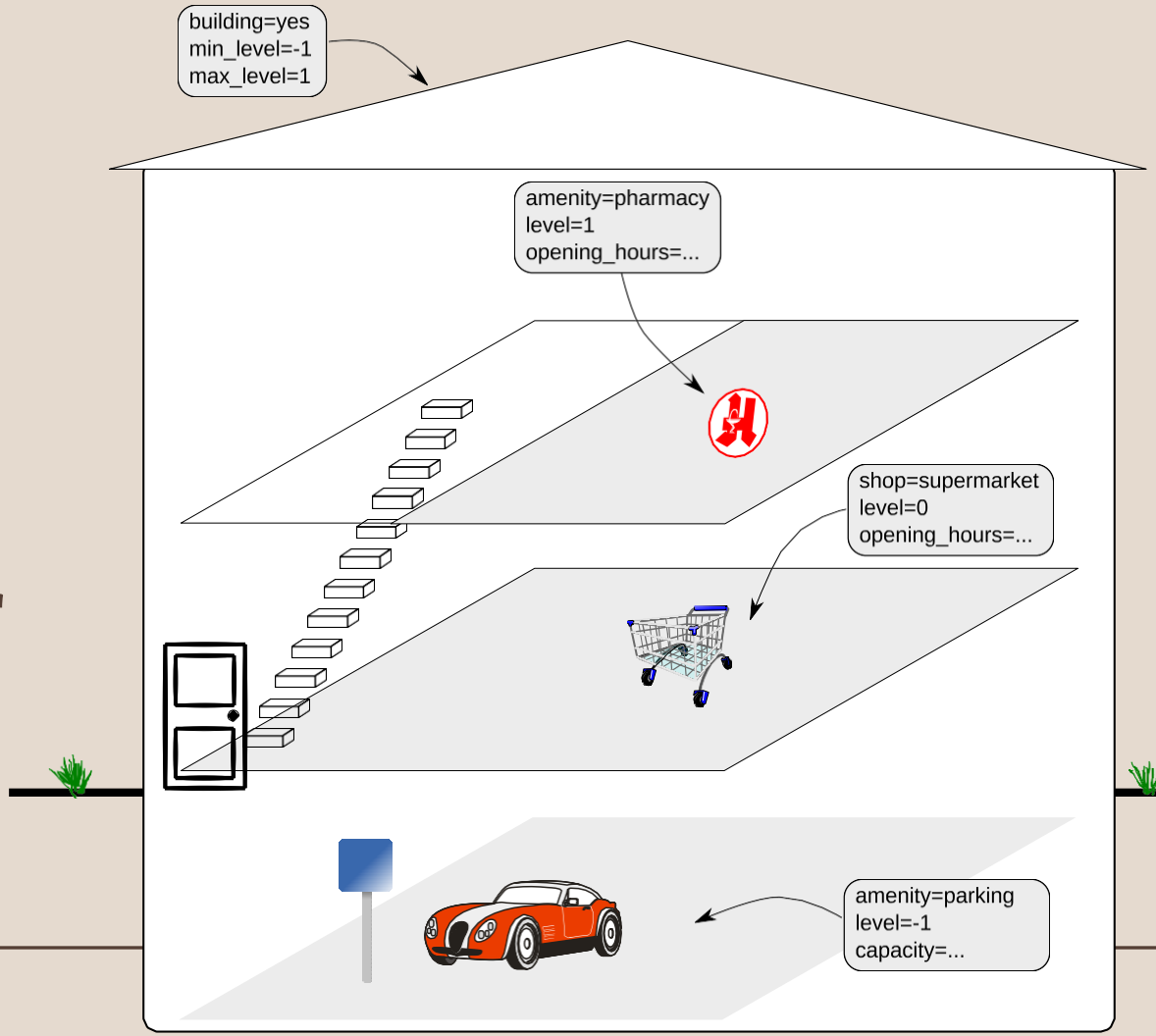
building=yes
min_level=-1
max_level=1



OSM indoor mapping

Indoor elements must be tagged with a **level=*** tag

This indicates the floor level on which the element is located

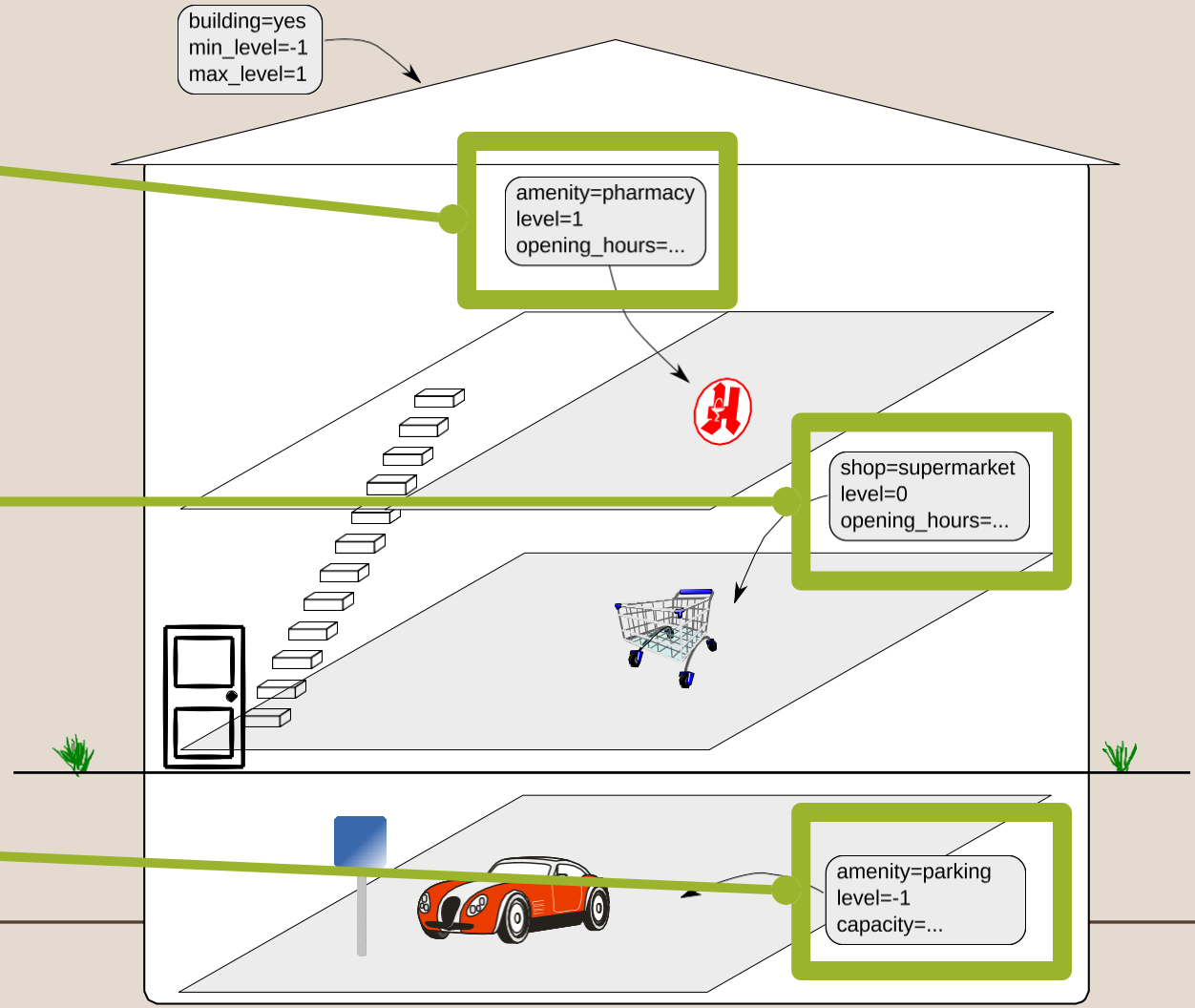


OSM indoor mapping

amenity=pharmacy
level=1

shop=supermarket
level=0

amenity=parking
level=-1



OSM indoor mapping

Indoor elements can be modelled as individual **nodes** (POIs)

adding the tag

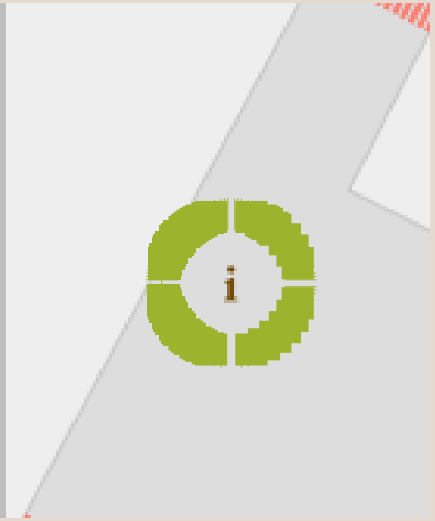
level=*

and the tag

indoor=yes

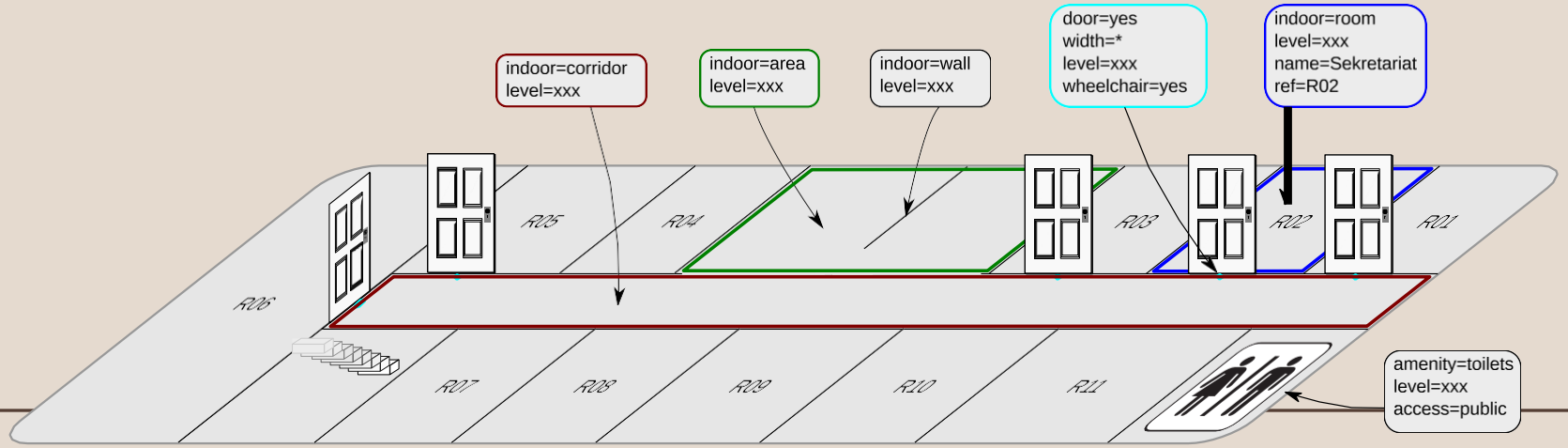
Tags

braille	yes
indoor	yes
information	tactile_map
level	1
tourism	information



OSM indoor mapping

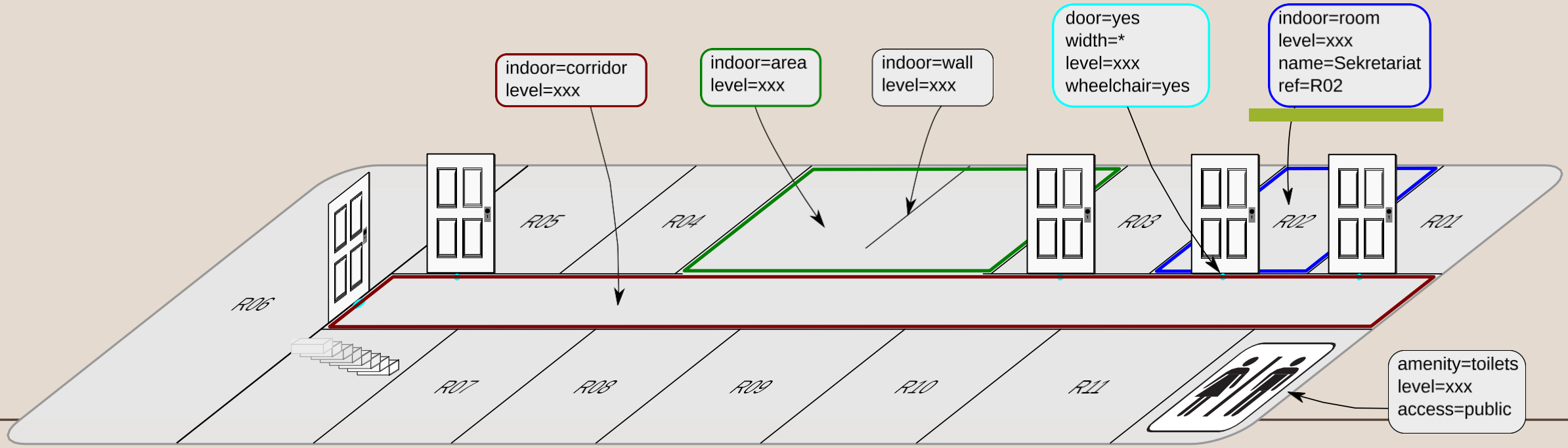
Indoor elements can be modelled
as **areas** (closed ways or multi-polygons)
The individual areas **share the same nodes**
where they share a wall in real world



OSM indoor mapping

The value of the **indoor** key specifies the basic indoor element

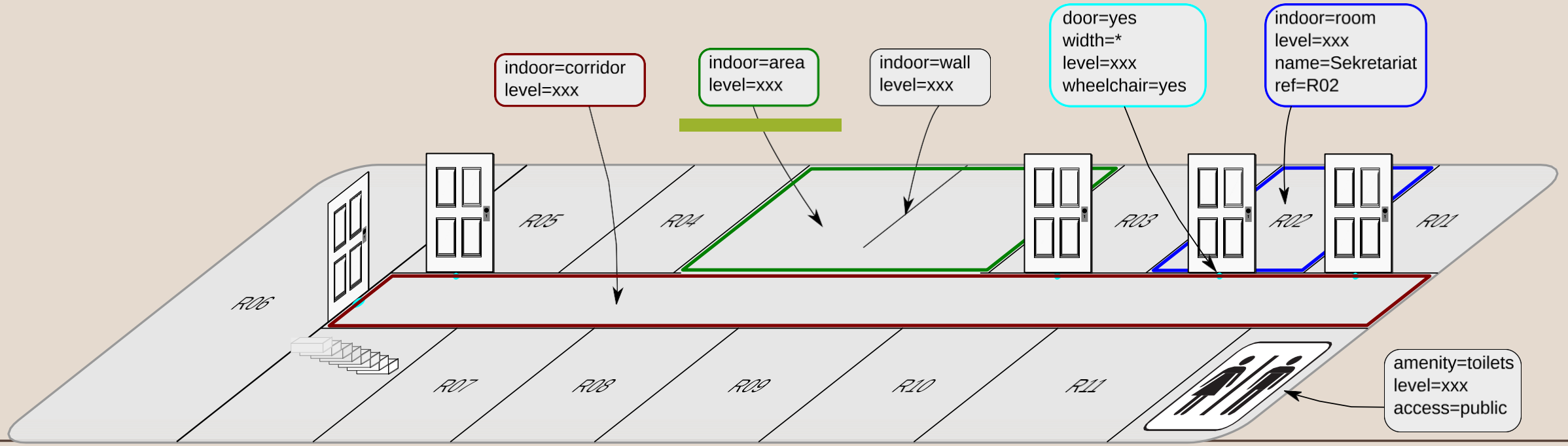
indoor=room > indicates a conventional room with walls The purpose of the room can be described by room=*



OSM indoor mapping

The value of the **indoor** key specifies the basic indoor element

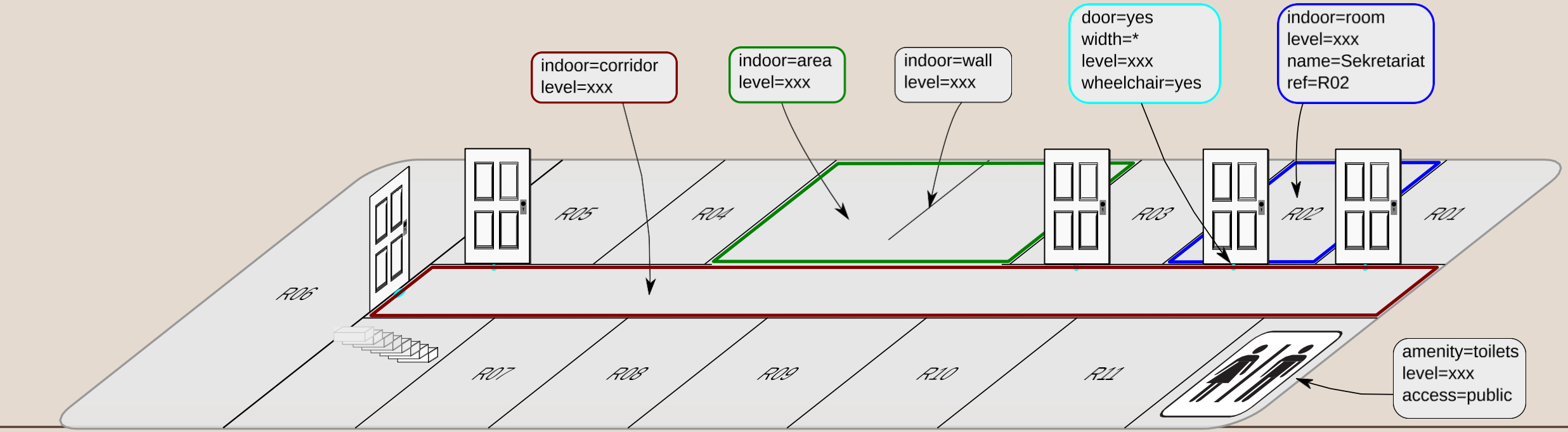
indoor=area > indicates an area without walls



OSM indoor mapping

The value of the **indoor** key specifies the basic indoor element

indoor=corridor > an enclosed walkway area that connects rooms



OSM indoor mapping

Connections between rooms are mapped with door nodes

A door node is shared between the rooms which the door connects

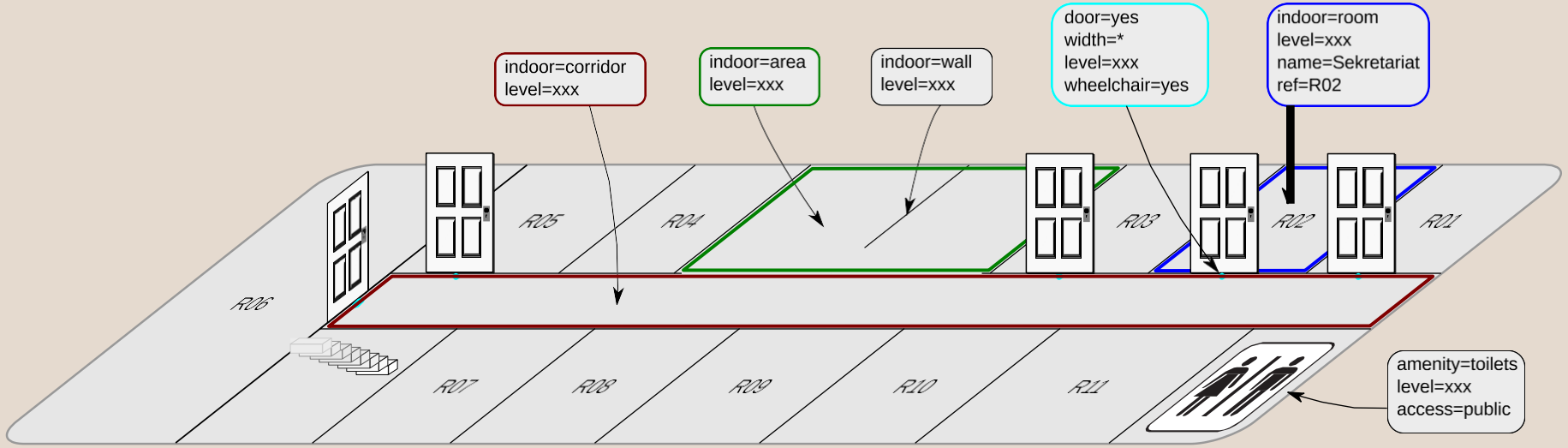
indoor=door

door=yes

level=*

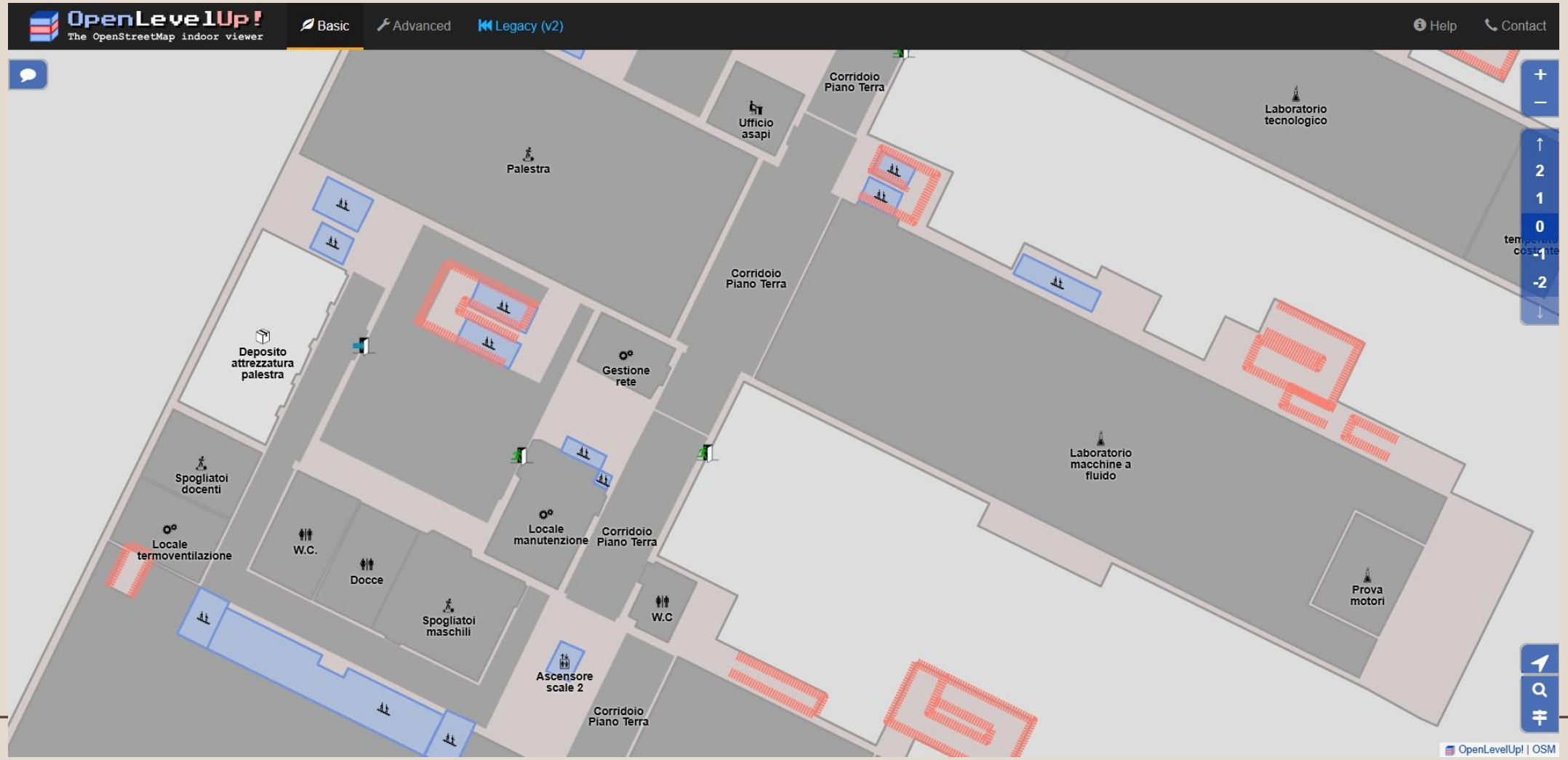
wheelchair=yes/no

access=yes/private



~~Doors which are entrances to or exits from the building are additionally tagged as entrance=*~~

Viewing OSM indoor data



Indoor elements to map

Classrooms (including study rooms and laboratories)

level=*

indoor=room

room=classroom/laborator

y name=*

capacity=*

access=yes

wheelchair=yes/no

~~Area surrounded by walls and doors~~

Indoor elements to map

Toilets

level=*

indoor=room

amenity=toilets

access=yes

wheelchair=yes/no

Area where all toilets are collected

Indoor elements to map

Reception

level=*

indoor=room

room=reception

access=private

Storage local

level=*

indoor=room

room=storage

access=private

Area surrounded by walls and doors

Indoor elements to map

Elevator

level=* {min:max}

indoor=room

highway=elevator

wheelchair=yes/no

Stairs

level=* {min:max}

indoor=room

stairs=yes

Area surrounded by walls and doors

Indoor elements to map

Corridor

level=*

indoor=corridor

Area surrounded by walls and doors

Indoor elements to map

Door

level=*

indoor=door

door=yes

wheelchair=yes/no

access=yes

Node shared between the rooms which the door connects

Indoor elements to map

Vending machine

level=*

indoor=yes

vending_machine=yes

vending=food

Defibrillator

level=*

indoor=yes

~~emergency=defibrillator~~

Fire extinguisher

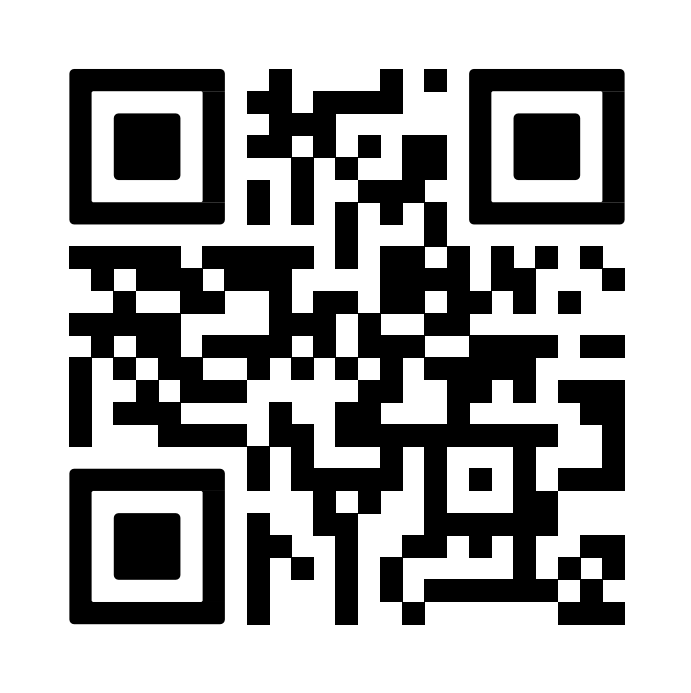
level=*

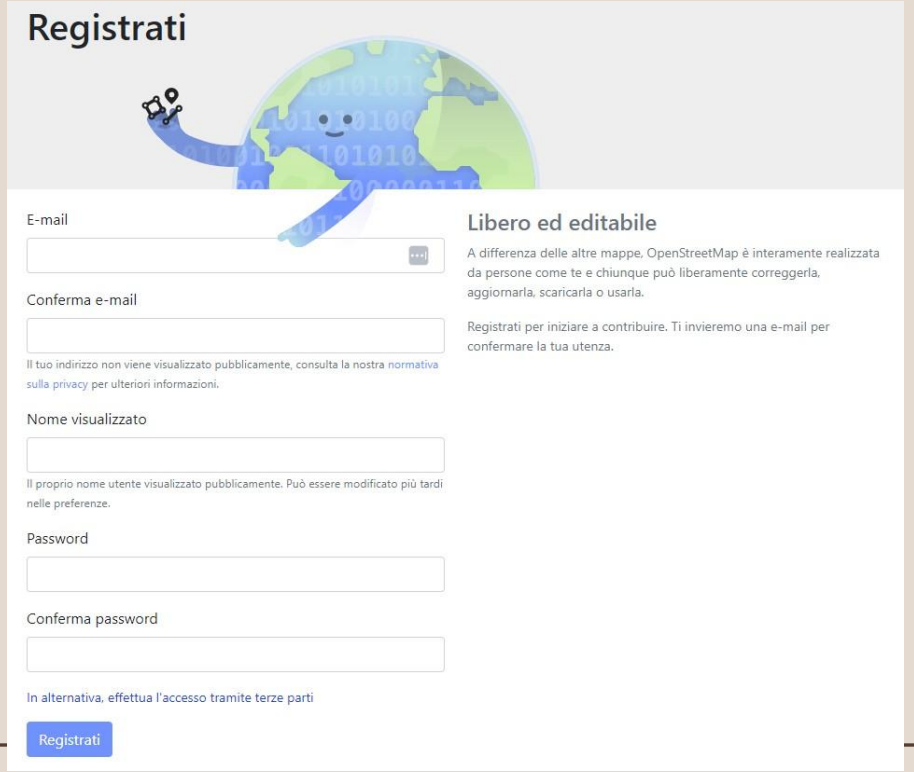
indoor=yes

emergency=fire_extinguisher

How to contribute

Creating an account on OpenStreetMap



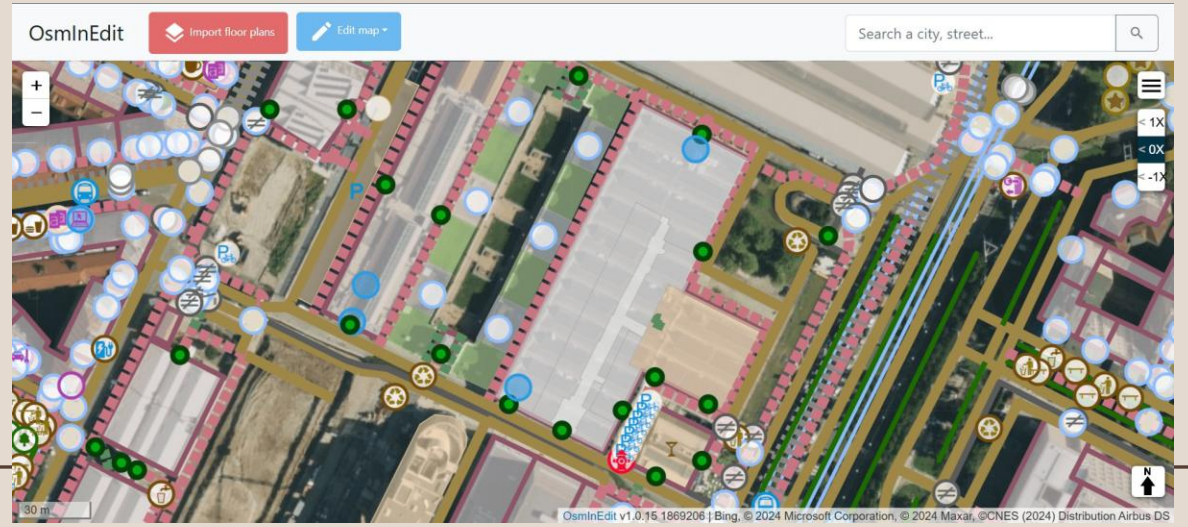


OsmInEdit

OsmInEdit (OpenStreetMap Indoor Editor)
is a web editor focused on indoor mapping

It offers **indoor-dedicated functionalities**

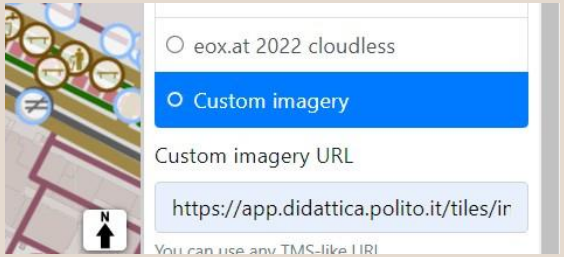
- Browse through all buildings and **levels**
- Import **images building plans** and **custom TMS** as background imagery
- Simple **editing of indoor data**: add or edit every single object in a **particular level** of a building with the support of **dedicated presets**: room, corridors, doors, furniture... and all common features (amenities, shops, offices...)



OsmInEdit

Add the Tile Map Service of the Politecnico floor plans as background in OsmInEdit

Click on the burger button and scroll down to Custom imagery



Insert the following URL for the **ground floor plan**

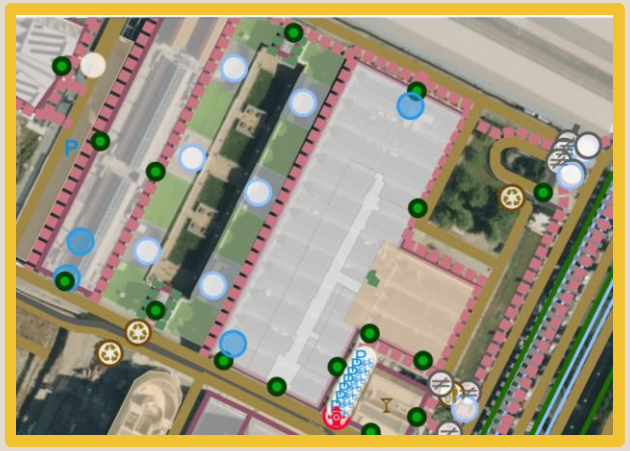
https://app.didattica.polito.it/tiles/int-light-xpte/{z}/{x}/{y}.png

Insert the following URL for the **basement floor plan**

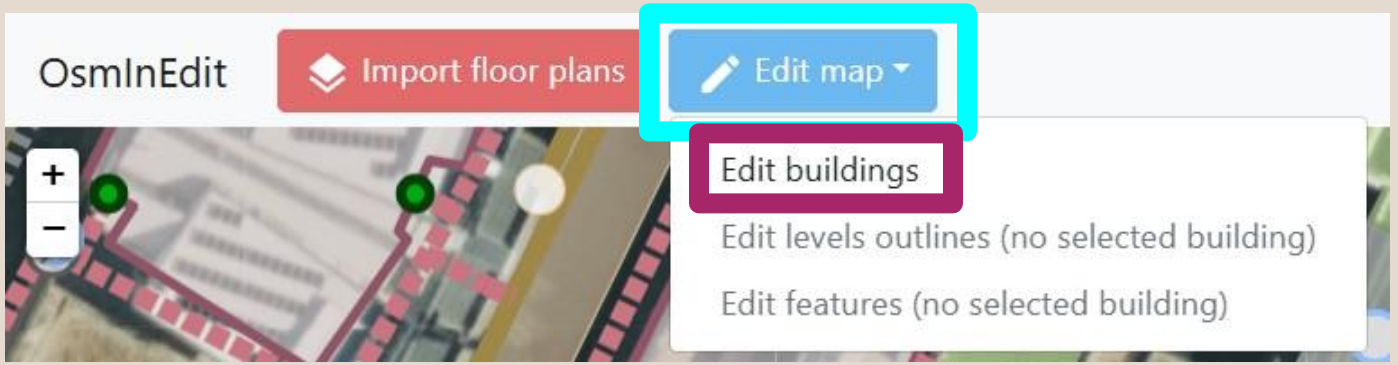
https://app.didattica.polito.it/tiles/int-light-xs01/{z}/{x}/{y}.png

OsmInEdit

Add a room



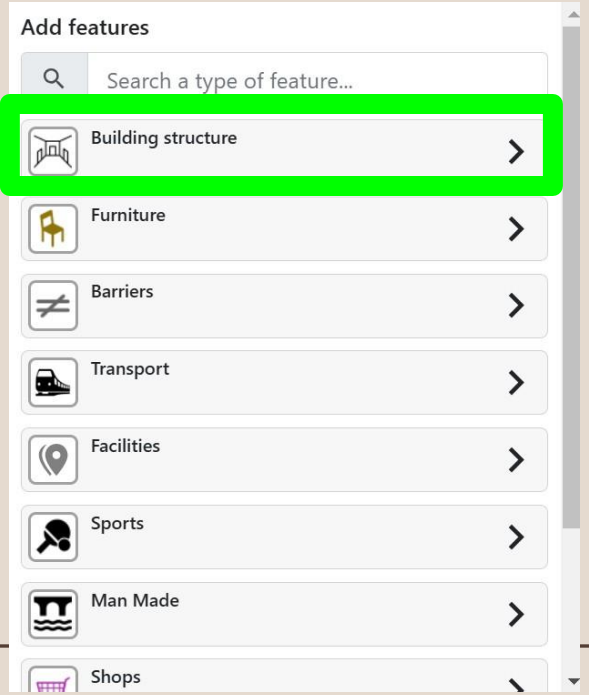
1. **Center the map** on the Politecnico area
2. Explore the **data filter by floor number** (on the right) and choose the **floor level** that you want to edit (e.g. 0)
3. Click on **Edit Map button** and choose **Edit buildings**



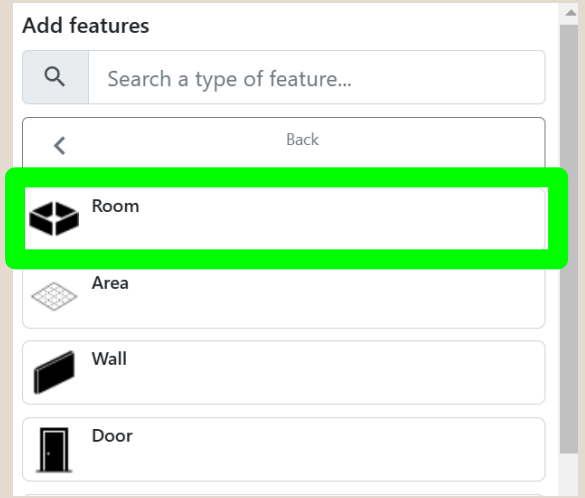
OsmInEdit

Add a room

1. **Select a building** by clicking on it
2. Click on **Edit features** button



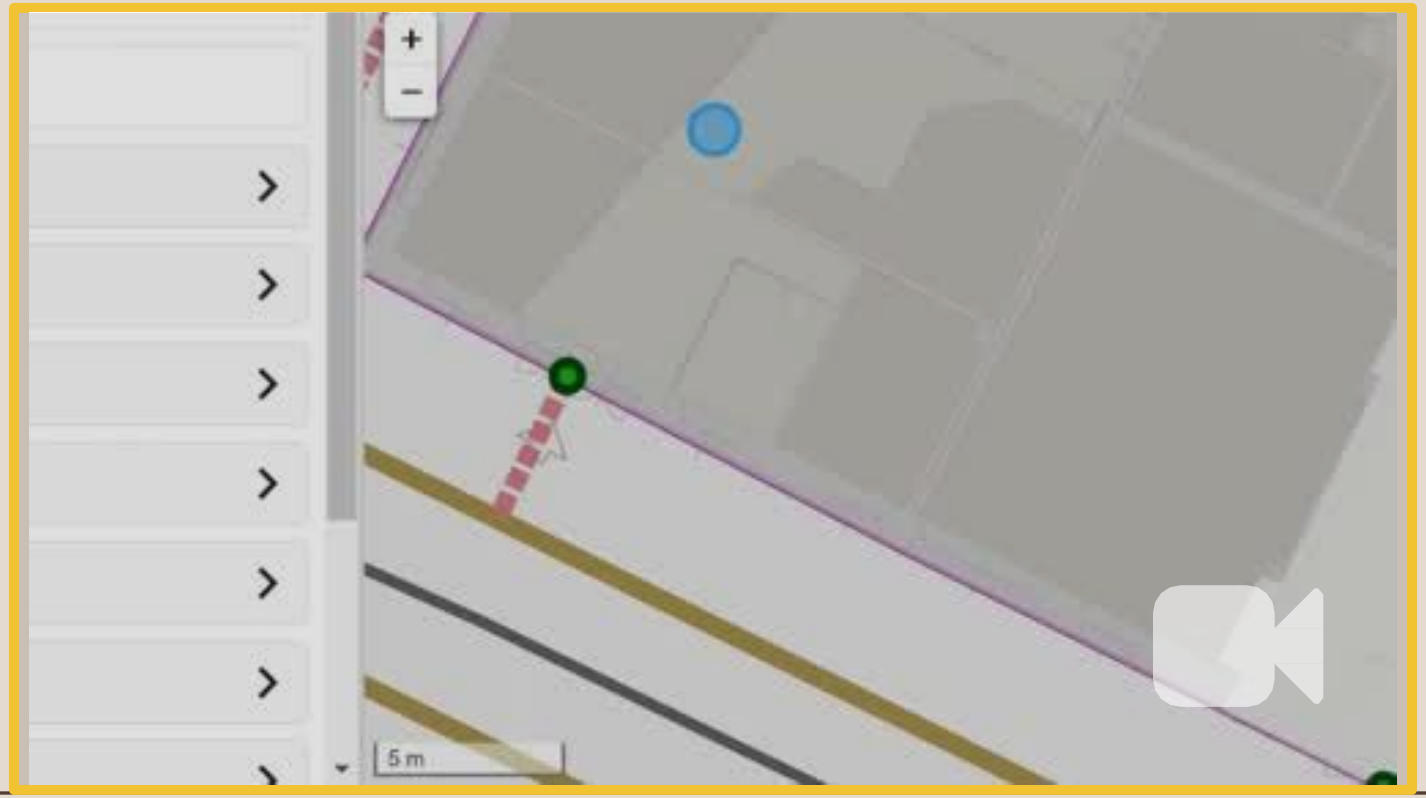
3. In the left panel, search for the element that you want to add: **Building structure --> Room**



OsmInEdit

Add a room

4. **Draw the outline of the classroom**
5. Pay attention to **snap** the vertices on **building walls** and on existing **indoor areas** (as corridor)



OsmInEdit

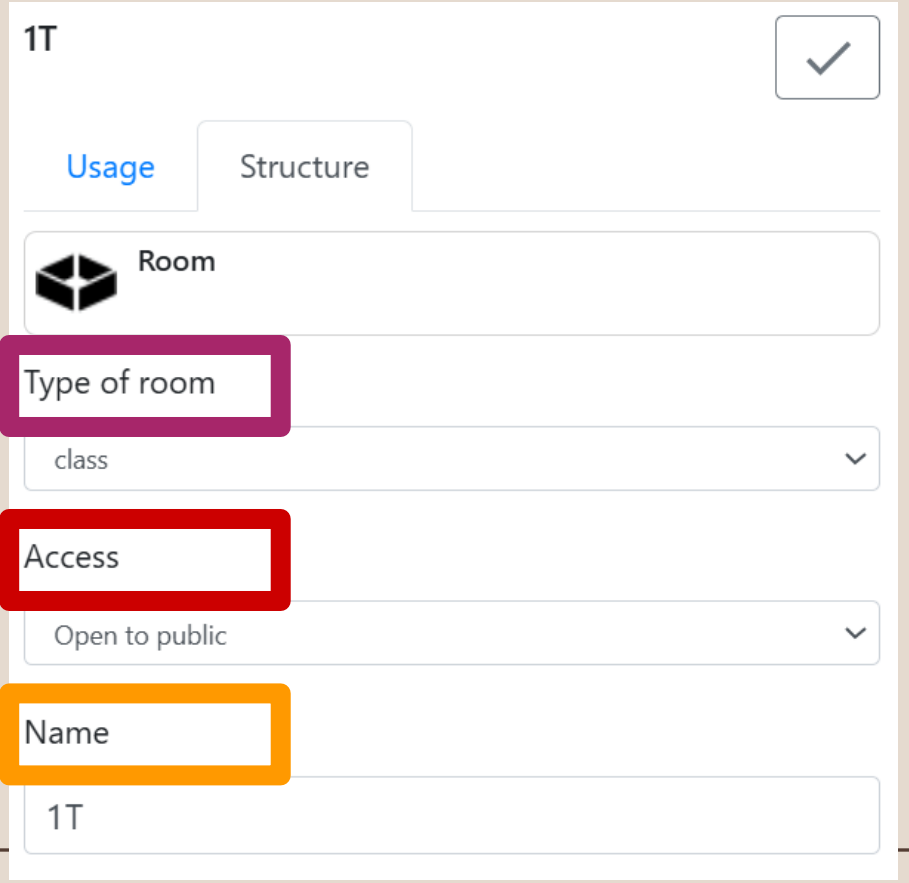
Add a room

6. In the left panel set the fields

   Type of room as class

   Access as Open to public

   Name as the name of the classroom

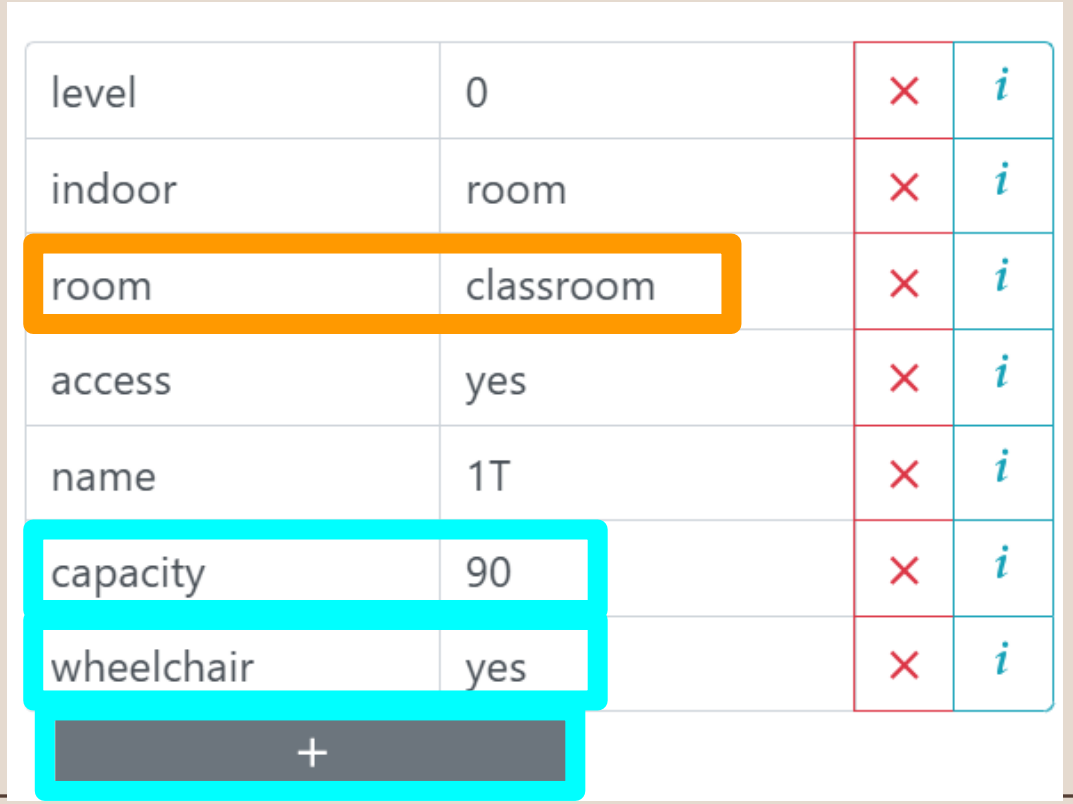


OsmInEdit

Add a room

6. Scroll down to see the list of the tags
   and **add** or **adjust values**
   add capacity key and set the value
   add wheelchair key and set the value
   adjust the value **from class to classroom**

7. Click on **Done** button
   on the top of the panel

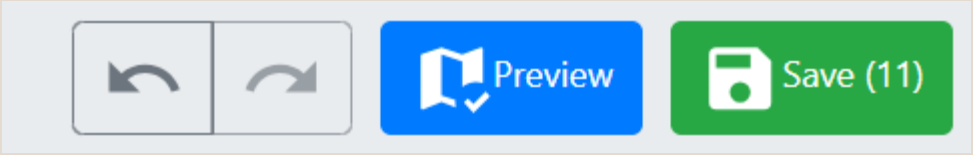


OsmInEdit

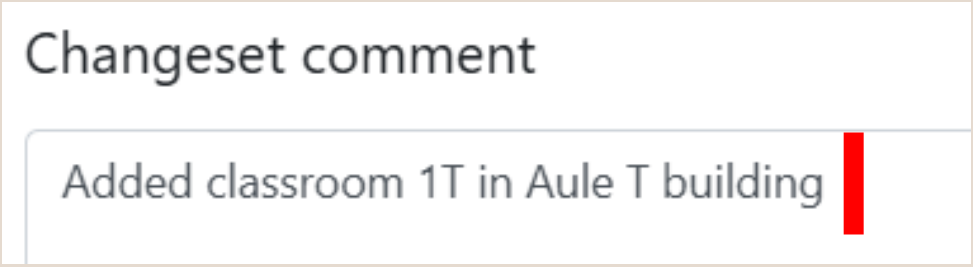
Save and upload youredits

It is possible to save multiple objects at once, but for today we will save **one object at a time**

8. Click on **Save** button



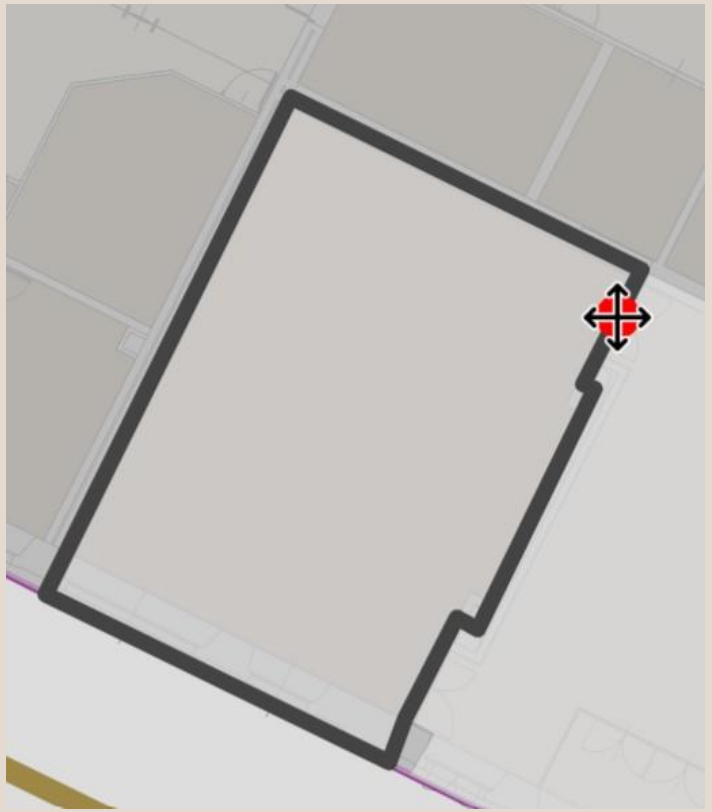
9. Write a **changeset comment** to describe the editing operation carried out



OsmInEdit

Add door

1. Select a building by clicking on it
2. Click on Edit features button
3. In the left panel, search for the element that you want to add: **Building structure -->Door**
4. Pay attention to **snap the point on room border**
5. Set relevant **tags**



OsmInEdit

Add other elements

Toilets	Facilities --> Facilities --> Toilets/Restrooms
Reception	Building structure --> Room
Storage local	Building structure --> Room
Elevator	Building structure --> Elevator
Stairs	Building structure --> Staircase
Vending machine	Shops --> Vending machine
Fire extinguisher	Facilities --> Emergency --> Fire extinguisher
Defibrillator	Facilities --> Emergency --> Automatic defibrillator
Bicycle parking	Transport --> Bicycle --> Parking
Water point	Facilities --> Facilities --> Drinking water
Bench	Facilities --> Facilities --> Bench

How to contribute

ID Editor

https://learnosm.org/en/beginner/