



# **SPATIAL (LINKED) DATA**

HOW TO HANDLE LOCATION

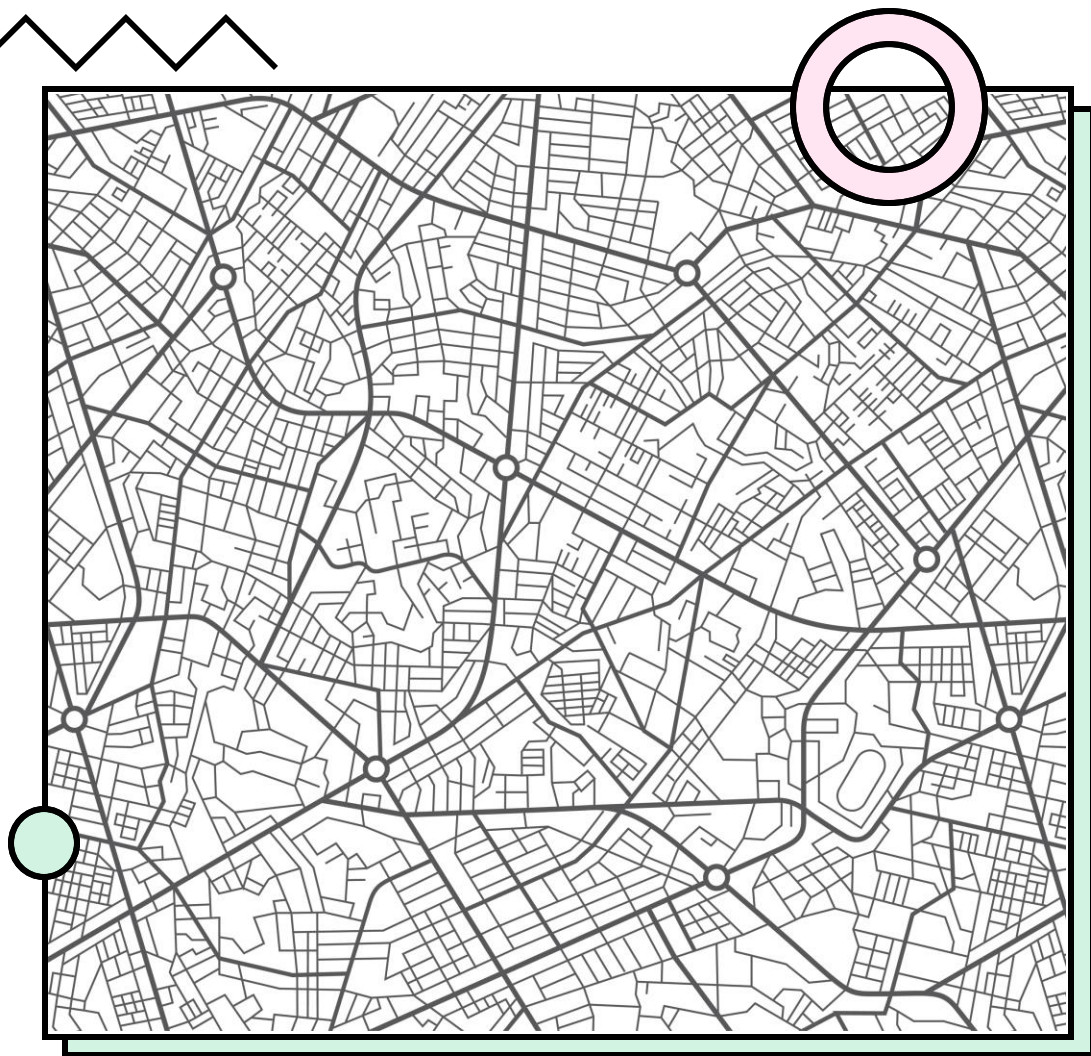
MICHAL MED

MICHAL.MED@FEL.CVUT.CZ





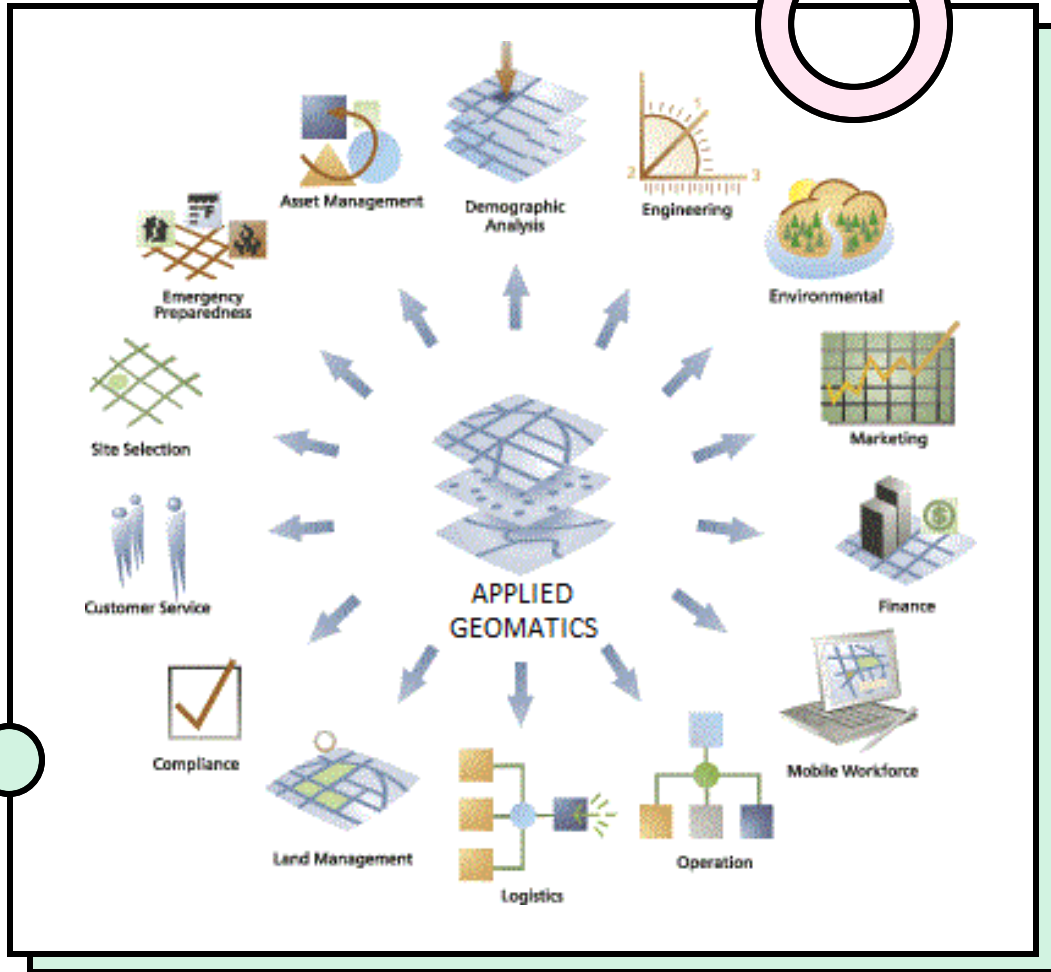
# WHAT IS SPATIAL



## Spatial data questions

- How far is it ...
- Which way to take?
- Where is the highest mountain?
- Which bus stops are reachable on foot in 5 minutes from my home?
- Who has the largest farm?
- What historical monuments are visible from my hotel room?





## Terminology

- Geographic Information Systems
- Geoinformatics
- **Geomatics**

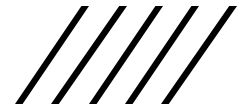
ISO/TC 211

*Discipline concerned with collection, distribution, storage, analysis, processing, presentation of geographic data or geographic information.*

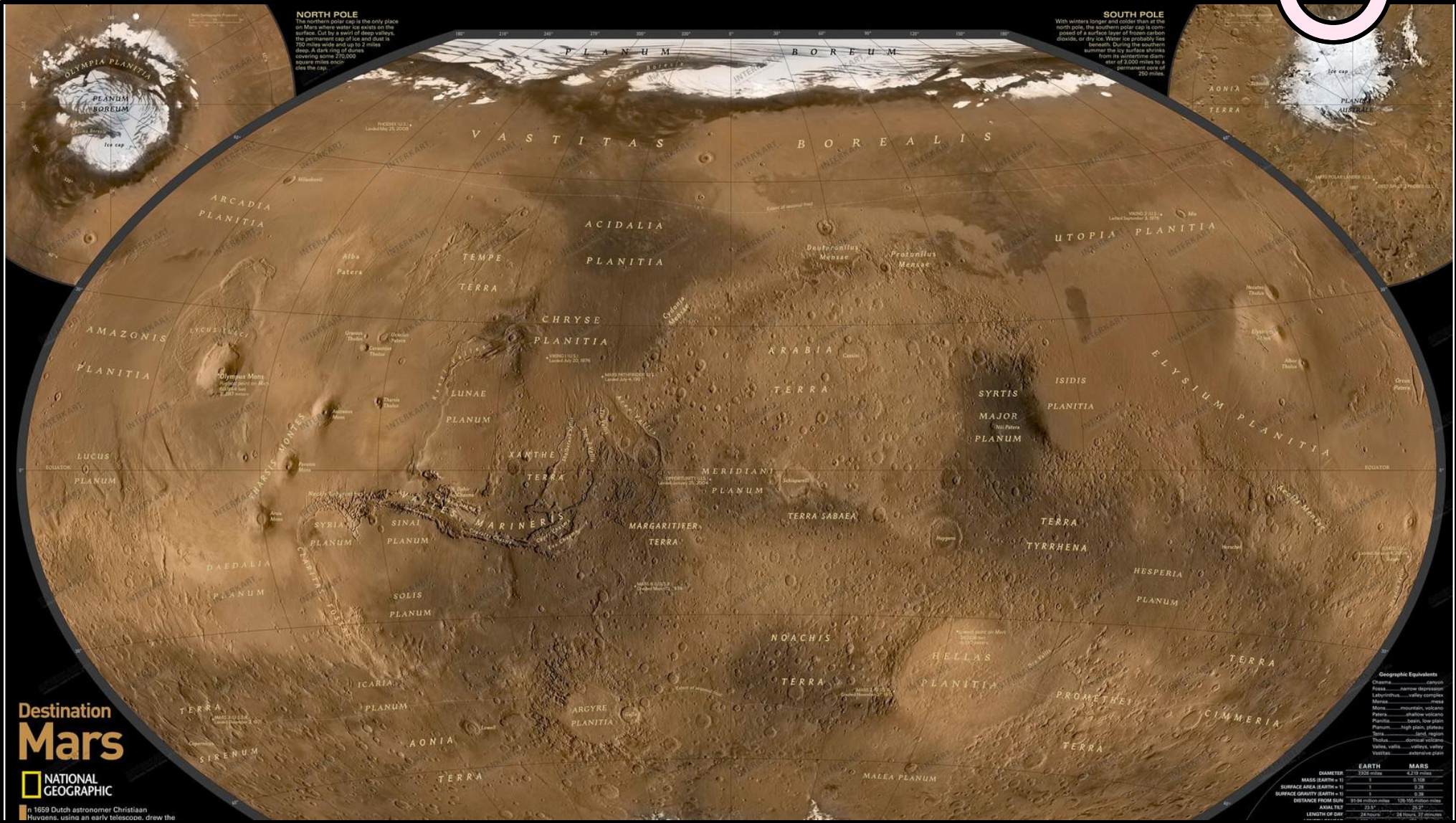
- **Geographic data**
- Geodata
- Spatial data
- Geospatial data and information
- Georeferenced data

ISO/TC 211

*Data and information having implicit or explicit association with a location relative to the Earth.*



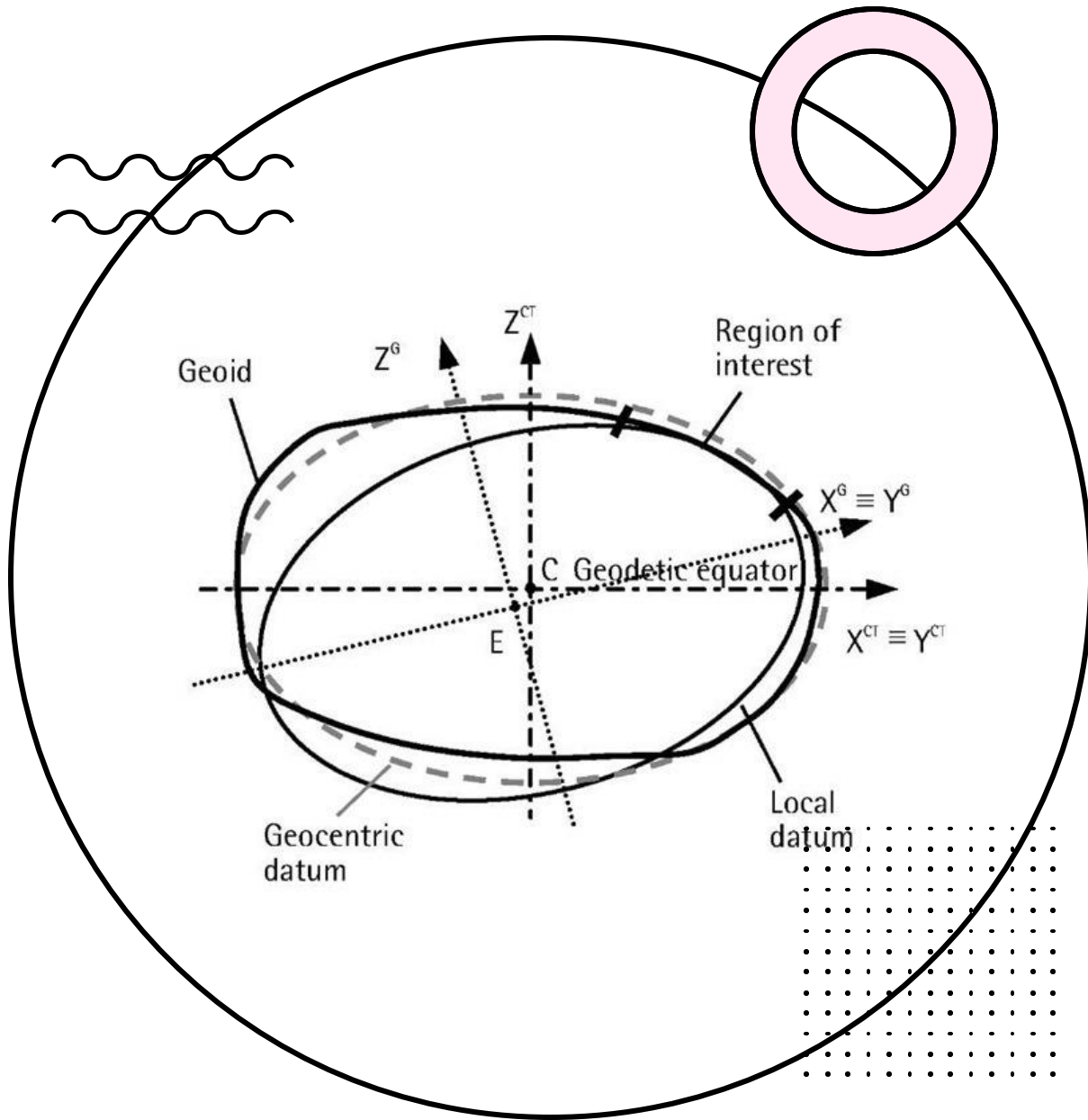




# Destination Mars



In 1659 Dutch astronomer Christiaan Huygens, using an early telescope, drew the



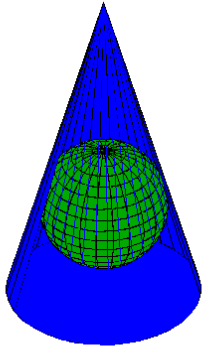
## Coordinate Reference System

Reference ellipsoid

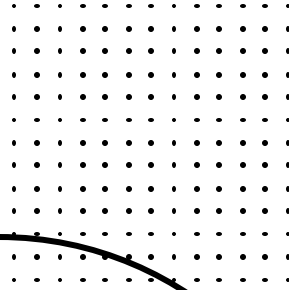
+

Datum

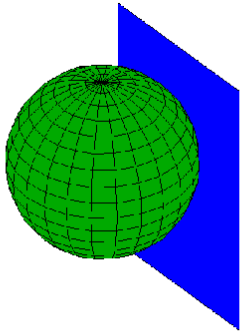
Peter H. Dana 9/20/04



Conical Projection Surface

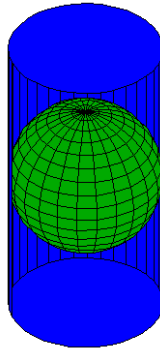


Peter H. Dana 9/20/04



Planar Projection Surface

Peter H. Dana 9/20/04



Cylindrical Projection Surface

- **Coordinate Reference System**

- + Projection

- WGS-84

- S-JTSK

- ETRS-89

- <http://epsg.io>

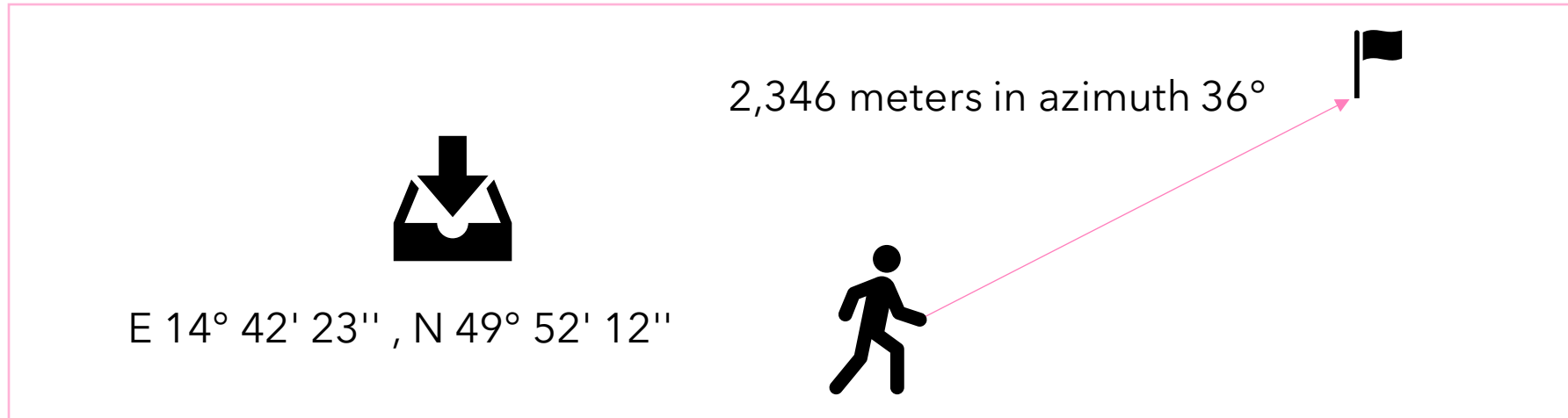


# WHAT IS GEODATA

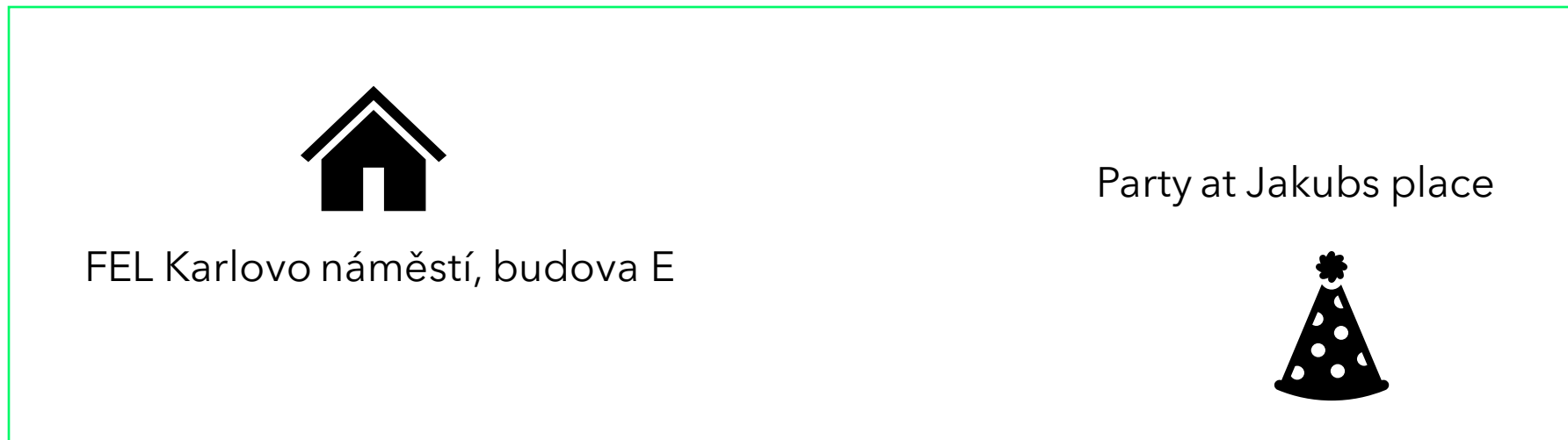




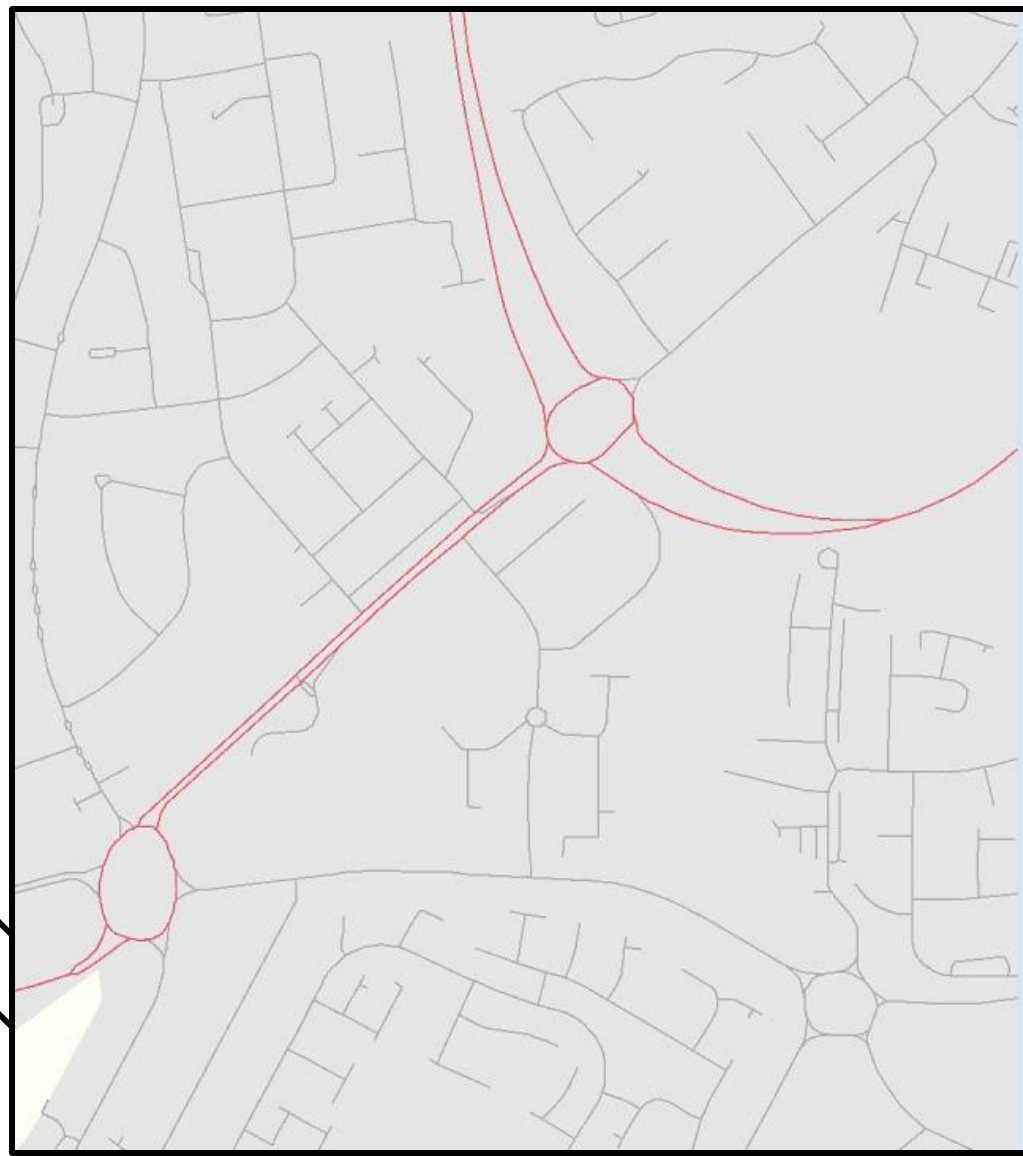
## Implicit geodata - coordinates, distances, directions



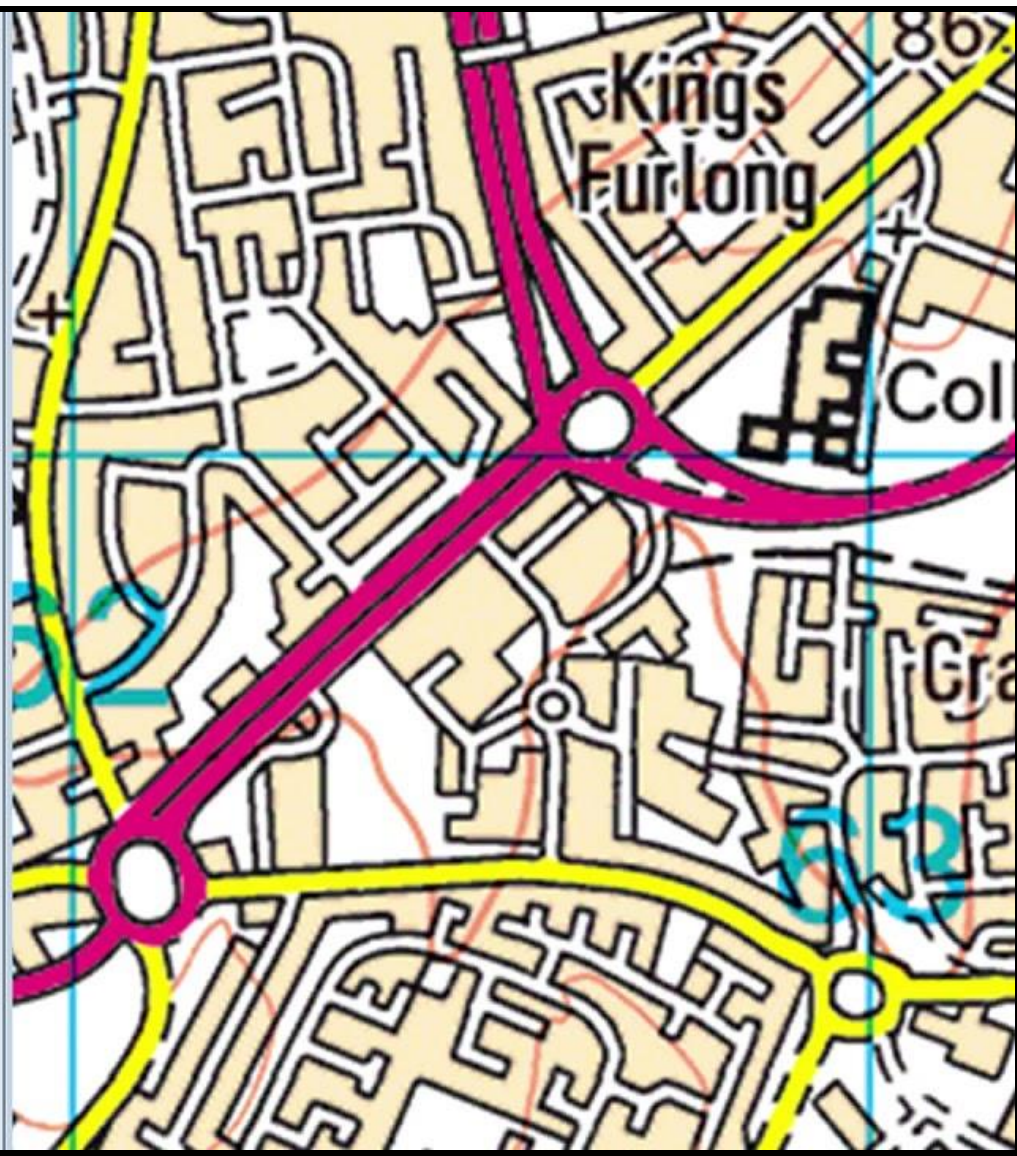
## Explicit geodata - reference, address, geographical name

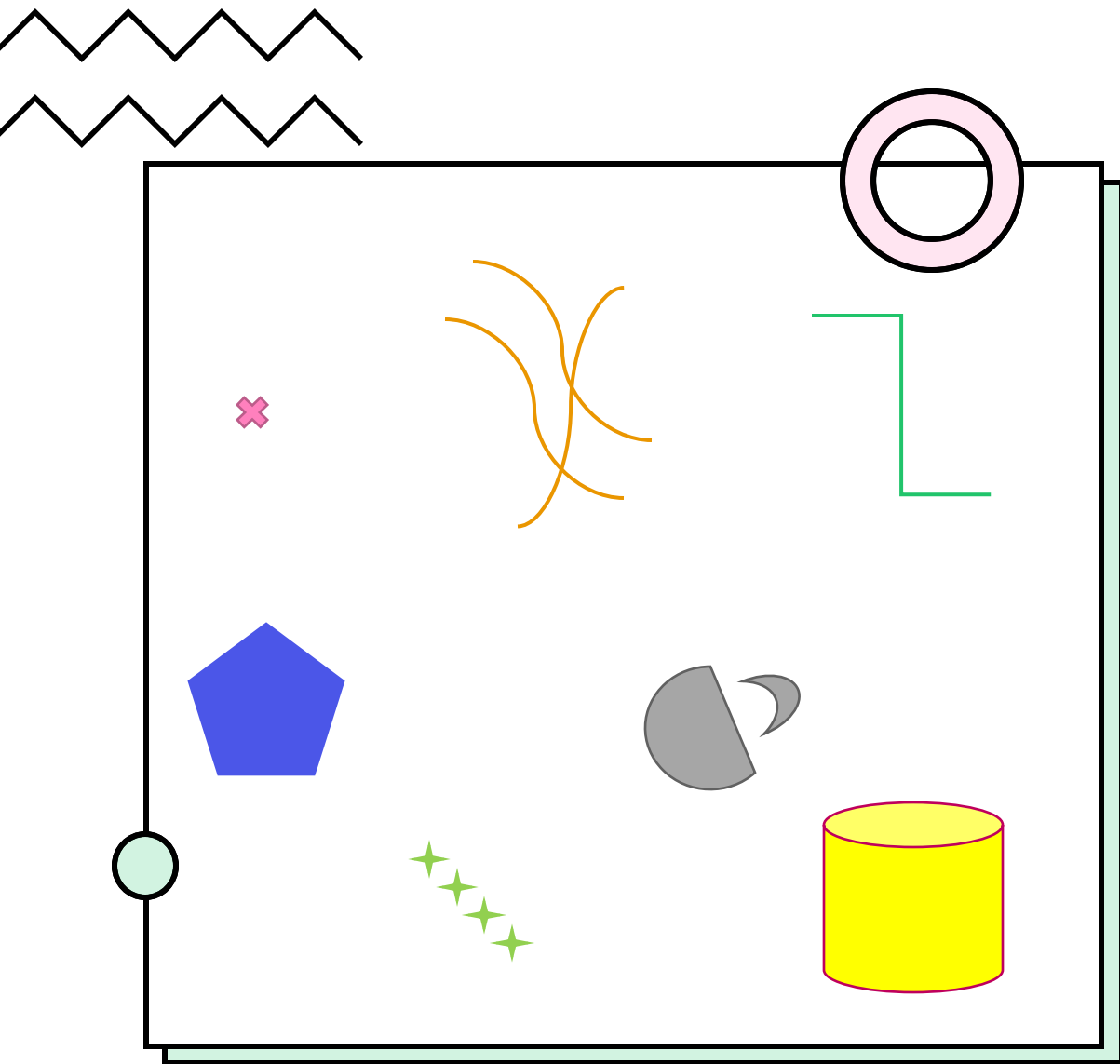


**Vector representation**



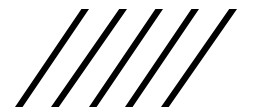
**Raster representation**





## Geometry objects

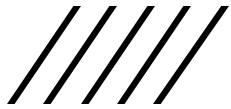
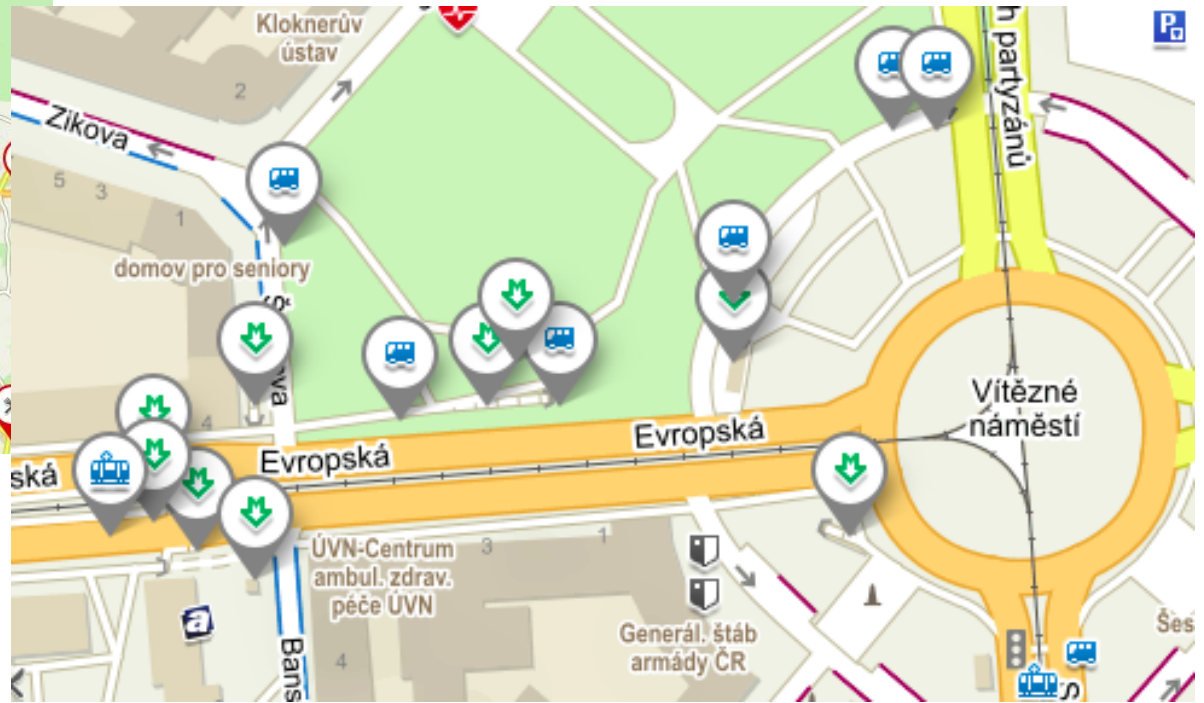
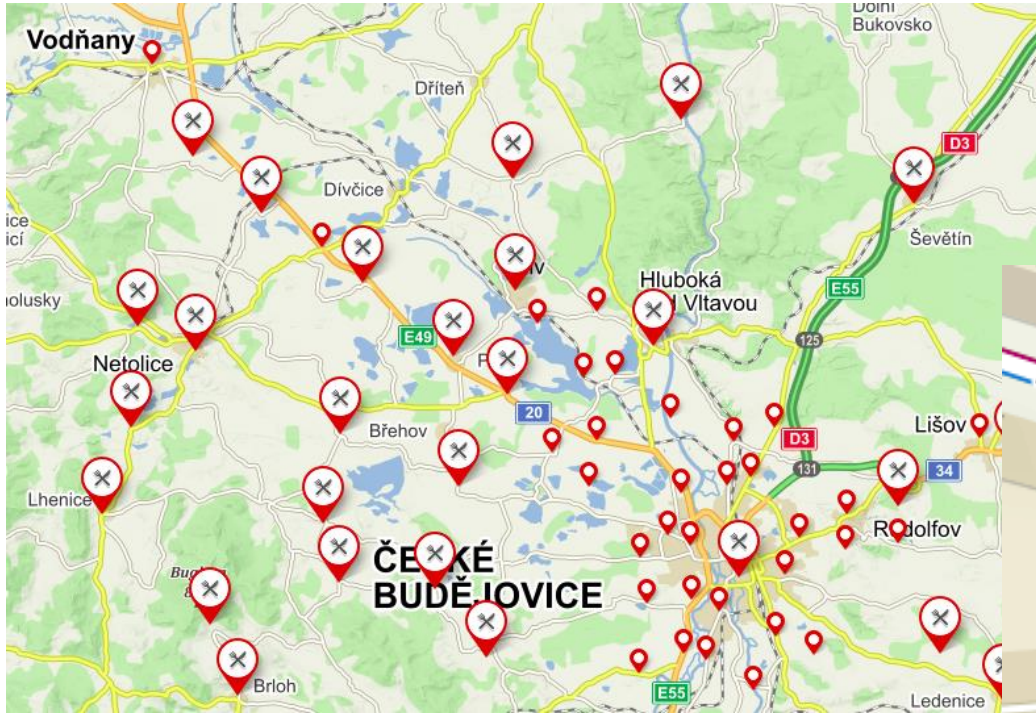
- Points
- Multipoints
- Lines (Linestrings)
- Multilines
- Polygons
- Multipolygons
- Surface
  
- w/o curves







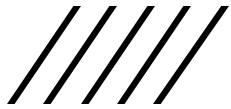
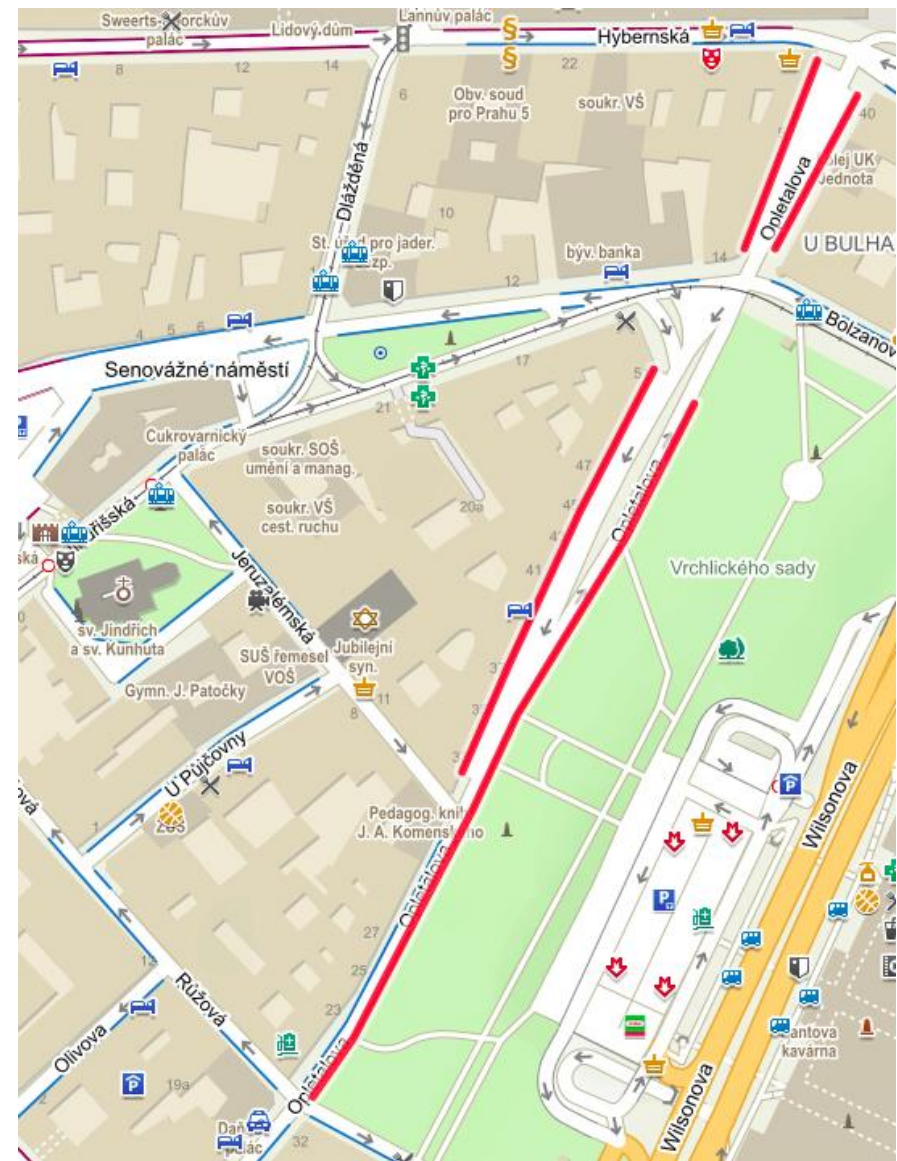
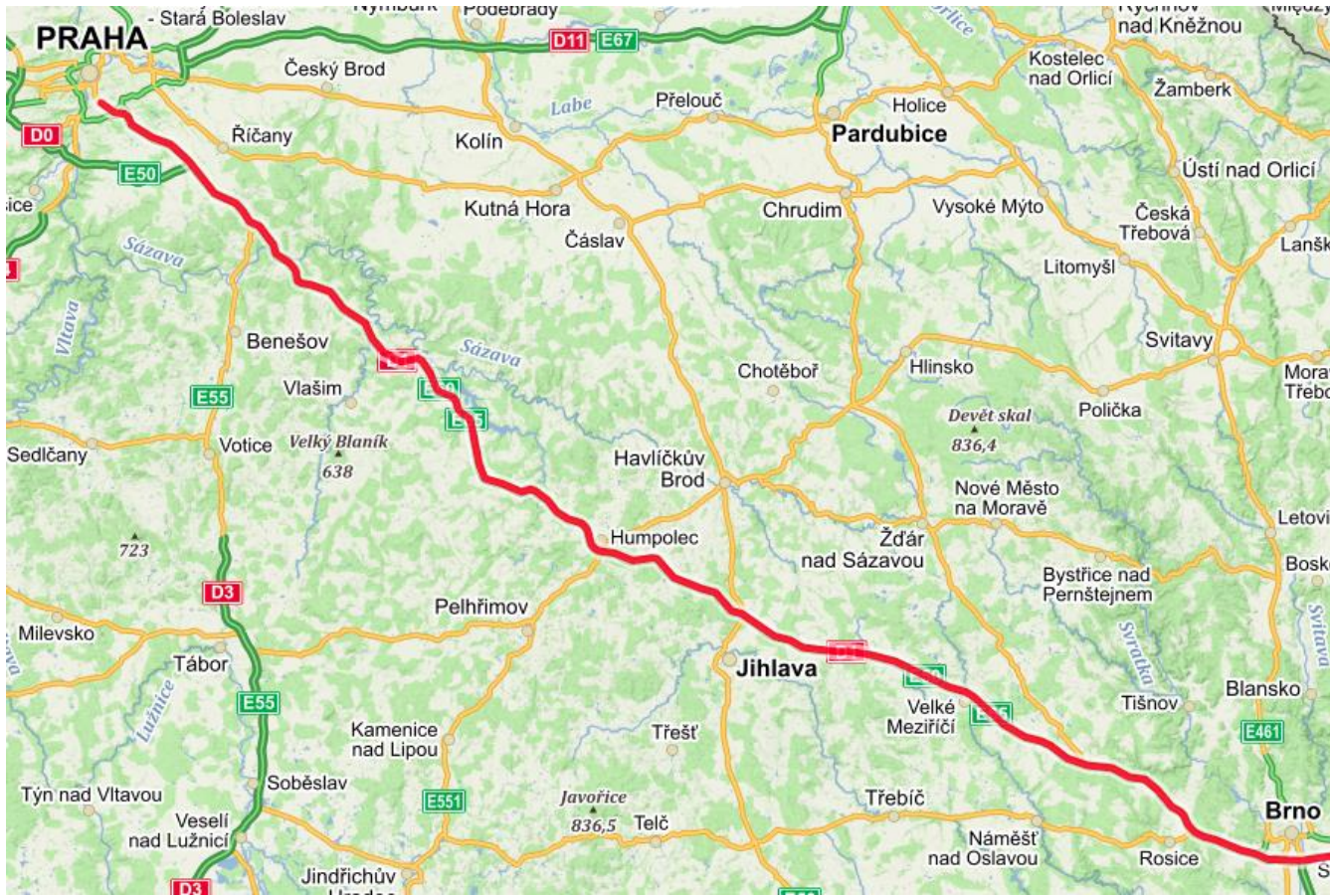
# Geometrical objects - points







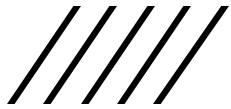
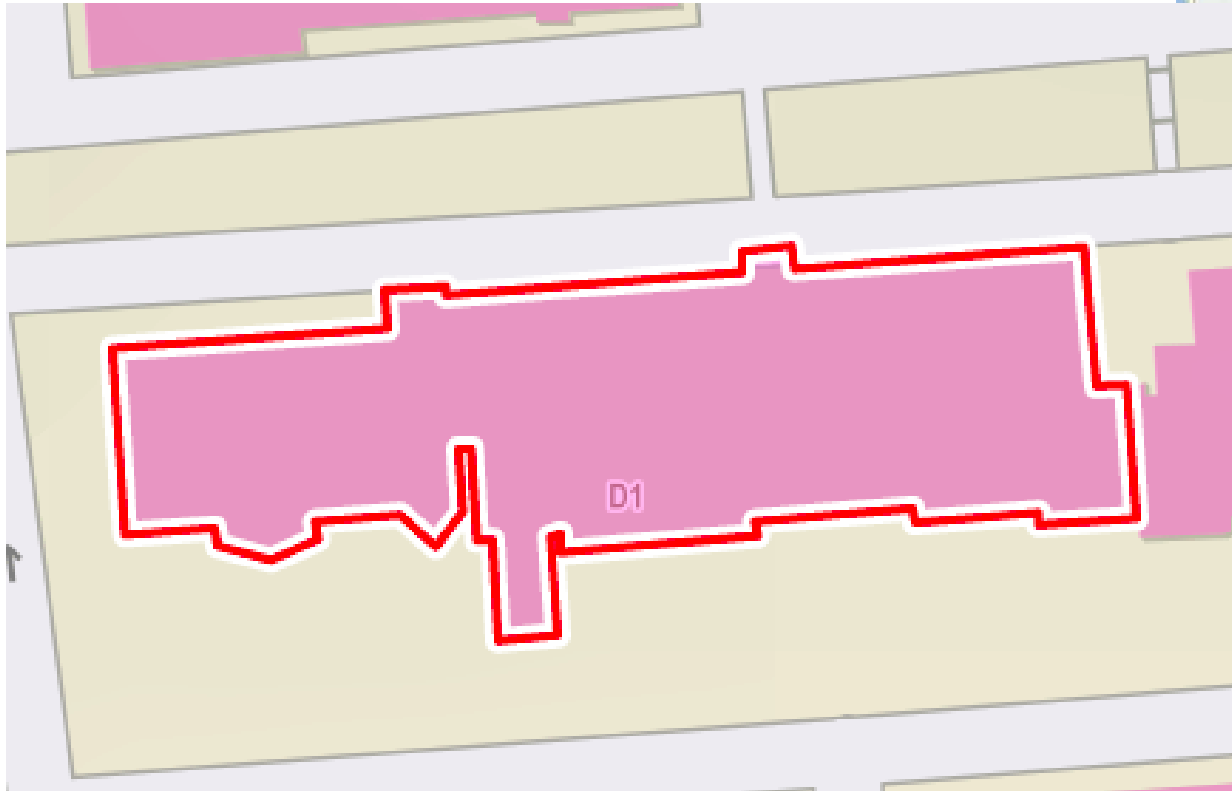
# Geometrical objects - lines





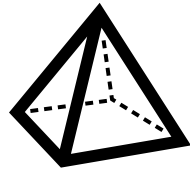


# Geometrical objects - polygons



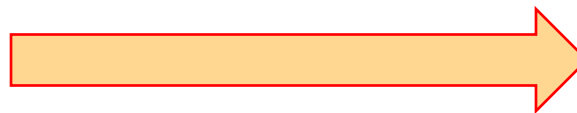
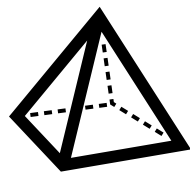
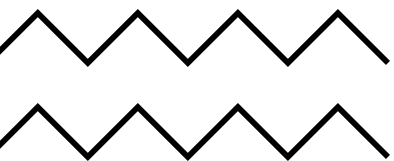
# Geometry representation

- how to represent geometry objects in data



## Geometry representation

- how to represent geometry objects in data

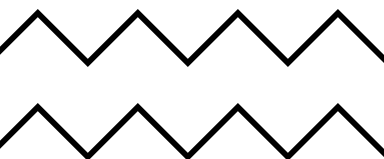


	X	Y
Point A	14.562°	49.8636°
Point B	14° 12' 56"	50° 02' 22"




## Geometry representation

- how to represent geometry objects in data



	X	Y
Point A	14.562°	49.8636°
Point B	14° 12' 56"	50° 02' 22"



Usually this is very format specific, which can make things very complicated.

Do not forget data representation depends on coordinate reference system and geometry object type.

How to specify coordinate reference system?

How to represent polygon? Multipolygon?



# Geometry representation - standard

## Well-Known Text (WKT)

```
POINT (50.056 14.434)
```

```
LINESTRING (50.056 14.434, 50.064 14.442, 50.042 14.445)
```

- OGC standard
- Specified in Simple Features Access (<https://www.opengeospatial.org/standards/sfa>) and ISO 19125
- Well described in [https://en.wikipedia.org/wiki/Well-known\\_text\\_representation\\_of\\_geometry](https://en.wikipedia.org/wiki/Well-known_text_representation_of_geometry)
- Suitable for representation of 2D objects
- Most libraries expects WGS-84, but WKT supports various CRS



# Geometry representation - standard

## Geography Markup Language (GML)

```
<gml:Point srsName="http://opengis.net/def/crs/EPSG/0/4326" srsDimension="2">/
  <gml:pos>50.056 14.434</gml:pos>
</gml:Point>

<gml:Curve srsName="http://opengis.net/def/crs/EPSG/0/5514" srsDimension="2">
  <gml:segments>
    <gml:LineStringSegment>
      <gml:posList>-641126.76 -1093821.18 -641119.35 -1093831.05
                  -641109.75 -1093844.44</gml:posList>
    </gml:LineStringSegment>
  </gml:segments>
</gml:Curve>
```

- Defined in OGC standard <https://www.ogc.org/standards/gml>
- Very robust, supports various CRS, various geometry objects, curves, 3D objects, coverage, sensor data
- The writing method is rather complicated (see above)



## Geometry representation - format based

Some spatial data formats may be using non-standardized way of geometry representation

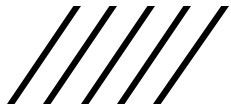
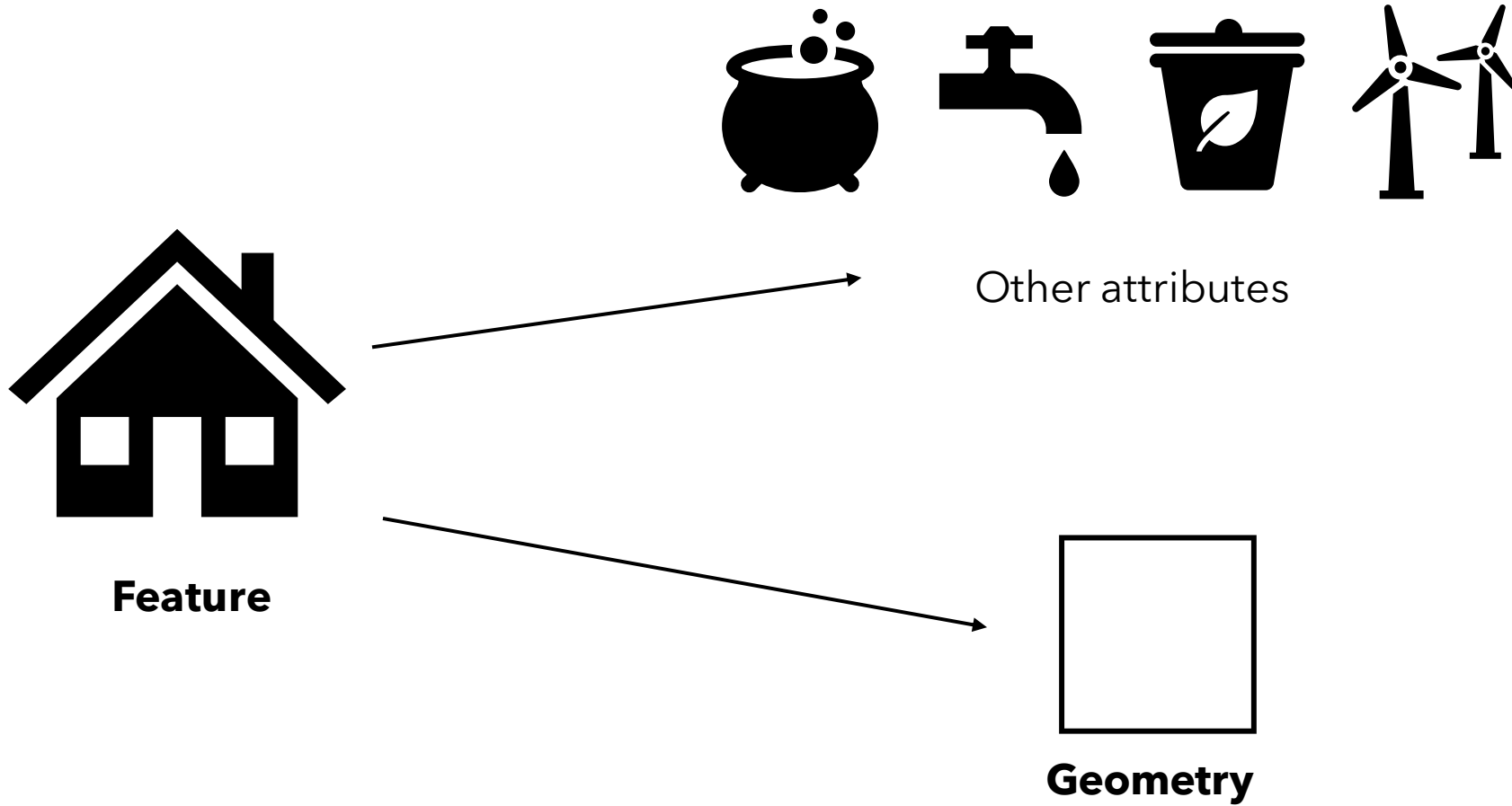
Data format	Geometry representation
GML	GML
GeoJSON	geojson
Shapefile	binary
GeoPackage	SQLite
CSV	any
GeoSPARQL	GML/WKT



# **SPATIAL DATA FORMATS**

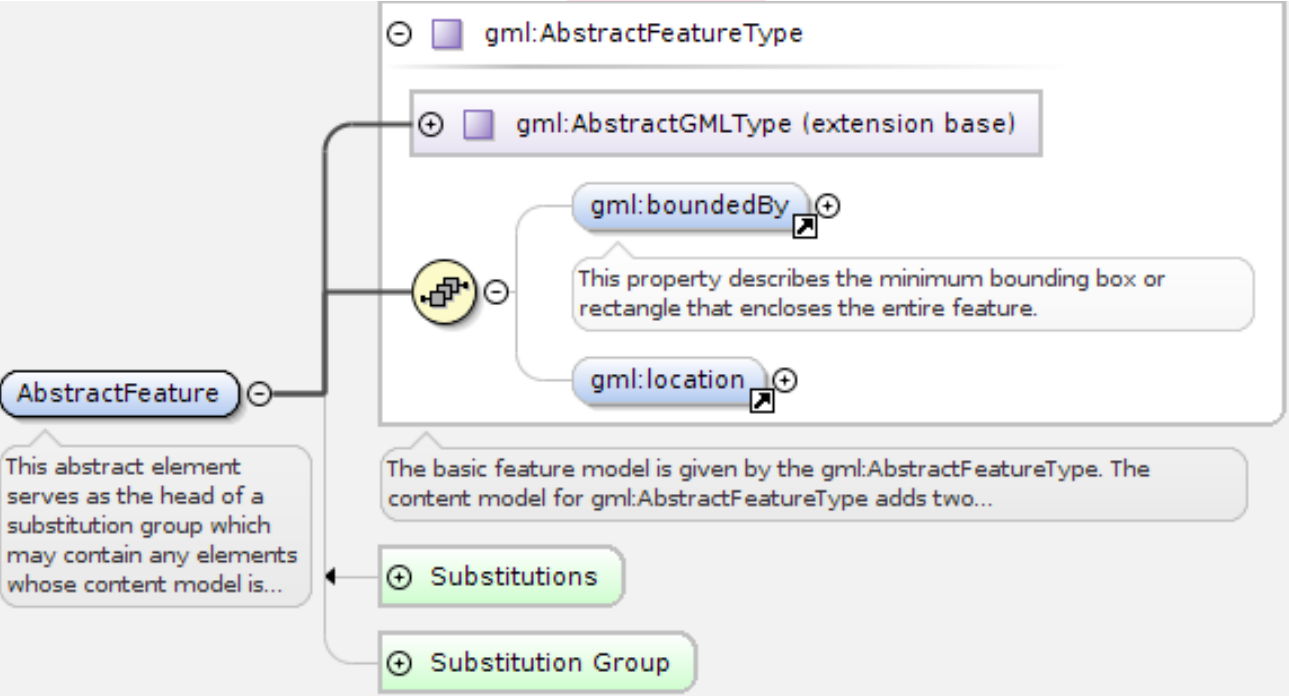


# Spatial data logic

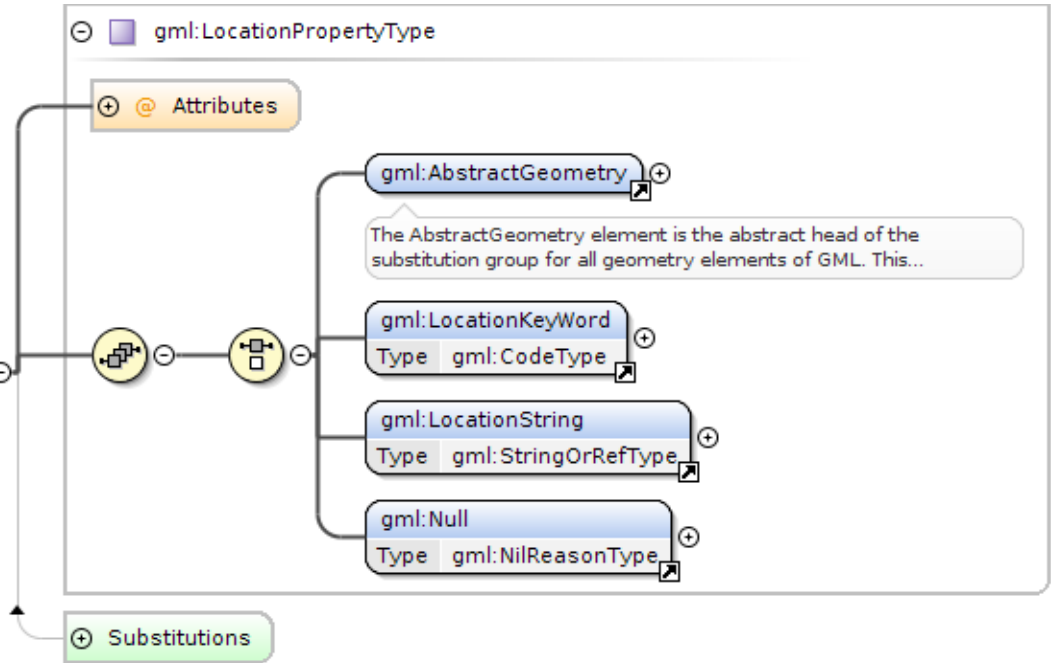


# Geography Markup Language (GML)

XML based format, described by set of XSD files available from: <http://schemas.opengis.net/gml/3.2.1/>



location  
Type gml:LocationPropertyType





# Geography Markup Language (GML)



```
<ad:Address gml:id="AD.22547665">
  <ad:inspireId>
    <base:Identifier>
      <base:localId>AD.22547665</base:localId>
      <base:namespace>CZ-00025712-CUZK_AD</base:namespace>
    </base:Identifier>
  </ad:inspireId>
  <ad:alternativeIdentifier>K Pitkovicům 1, Benice, 10300 Praha 10</ad:alternativeIdentifier>
  <ad:position>
    <ad:GeographicPosition>
      <ad:geometry>
        <gml:Point gml:id="P.AD.22547665" srsName="urn:ogc:def:crs:EPSG::5514" srsDimension="2">
          <gml:pos>-731037.56 -1053052.98</gml:pos>
        </gml:Point>
      </ad:geometry>
      <ad:specification xlink:href="http://inspire.ec.europa.eu/codelist/ GeometrySpecificationValue/entrance"
xlink:title="entrance"/>
      <ad:default>true</ad:default>
    </ad:GeographicPosition>
  </ad:position>
  <ad:component xlink:href="#AA.MOP.108" xlink:title="Praha 10"/>
  <ad:component xlink:href="#AA.MOMC.538078" xlink:title="Praha-Benice"/>
  <ad:component xlink:href="#AA.2585" xlink:title="Benice"/>
  <ad:component xlink:href="#TF.498211" xlink:title="K Pitkovicům"/>
  <ad:component xlink:href="#PD.10300" xlink:title="10300"/>
</ad:Address>
```



# Geography Markup Language (GML)

GML is used in the Infrastructure for Spatial Information in Europe (INSPIRE) as the main format for data. Regulation aims at sharing (spatial) data about environment accross Europe in a standardized way.



Data catalogue is available from <https://inspire-geoportal.ec.europa.eu/>

# GeoJSON

```
{
  "geometry": {
    "coordinates": [
      14.419134,
      50.090122
    ],
    "type": "Point"
  },
  "crs": {
    "type": "name",
    "properties": {
      "name": "urn:ogc:def:crs:EPSG::4326"
    }
  },
  "properties": {
    "cislo_orientacni": "22",
    "cislo_popisne": "128",
    "druh_mista": "RESTAURAČNÍ ZAHŘÁDKY",
    "druh_zbozi": "",
    "momc": "Praha 1",
    "ulice": "Pařížská"
  },
  "type": "Feature"
}
```

- JSON based format
- Own geometry representation
- Does not support other CRS than WGS-84 (functionality was removed)
- Geometry objects supported: Point, Multipoint, LineString, MultiLineString, Polygon, Multipolygon
- <http://geojson.io>
- Supported visualization in GitHub



# Shapefile



	JMENO	DATA50_K	DATA50_P
1	Jáchymka	6260000	vstup do je...
2	Netopýrka	6260000	vstup do je...
3	NULL	6260000	vstup do je...
4	Zbrašovské...	6260000	vstup do je...
5	Černotínsk...	6260000	vstup do je...
6	Výpustek	6260000	vstup do je...
7	Pekárna	6260000	vstup do je...
8	Švédův stůl	6260000	vstup do je...
9	Ochozská j...	6260000	vstup do je...
10	Cikánská	6260000	vstup do je...
11	U jezevce	6260000	vstup do je...
12	U žida	6260000	vstup do je...
13	Šámalíkovy...	6260000	vstup do je...
14	Šípka	6260000	vstup do je...
15	Ledové sluje	6260000	vstup do je...
16	Na Turoldu	6260000	vstup do je...

Show All Features

- Format created by ESRI company, but the format itself is (yet) open
- Native format for most used GIS in Czech Republic
- Consist of multiple files
- Restricted number of characters per column name
- Only one feature type per dataset
- Lot of known issues



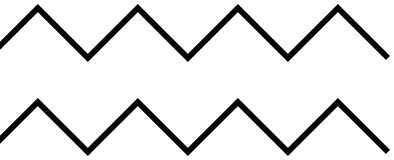
# OGC GeoPackage



- SQLite database file
- Supports simple and complex geometry structures (as an attribute)
- Supports both vector and raster data (in one file)
- Very fast and very complex

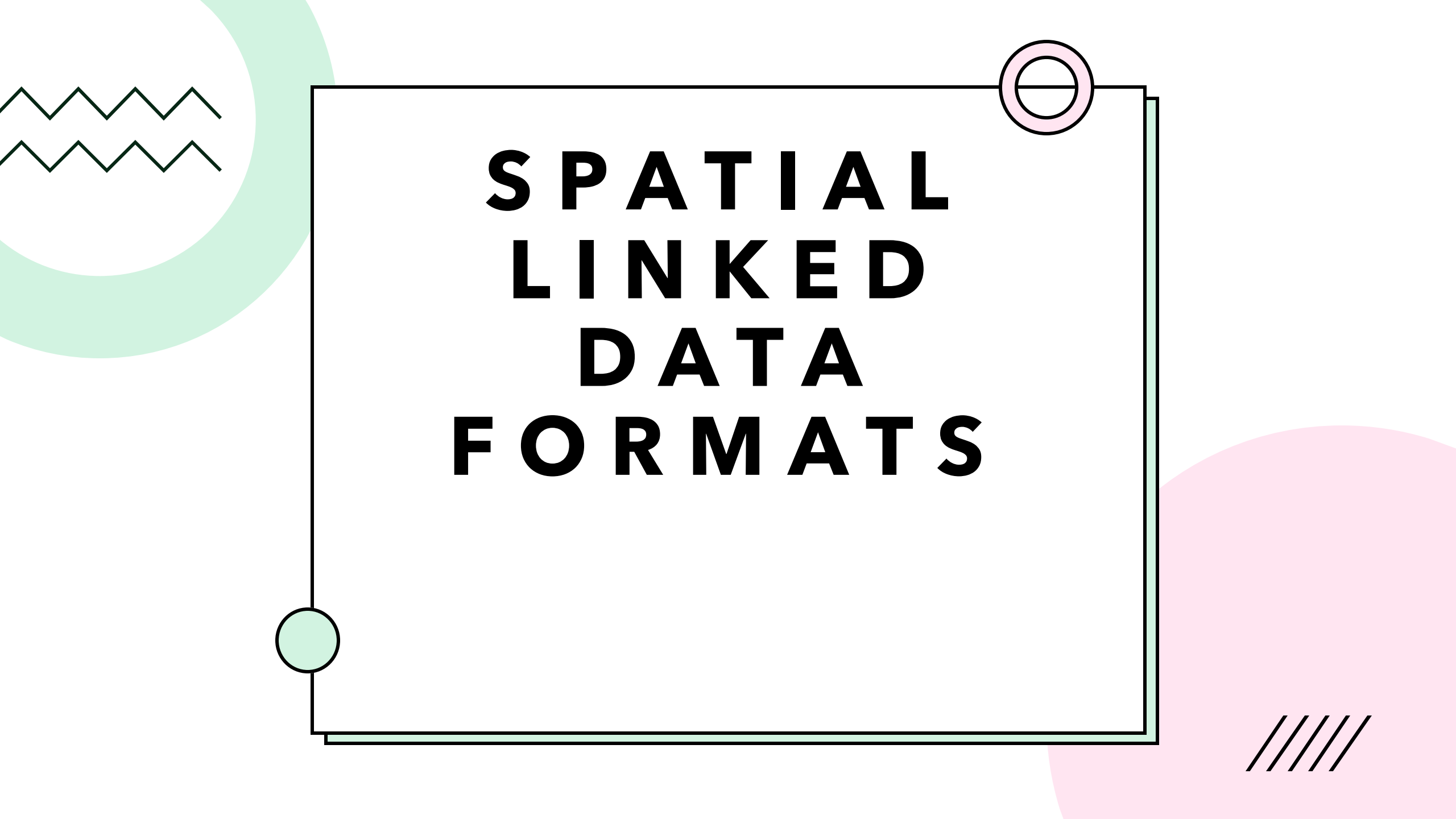


# Comma Separated Values



- Very easy
- MS Excel friendly
- Does not have recommended geometry (can be WKT)
- Geometry objects usually contain commas - must be escaped

```
1 Kód,Název ulice,Kód Obce,Název Obce,Kód Okresu,Název Okresu,WKT_Geometry,CRS
2 442666,Adamovská,554782,Praha,3100,Hlavní město
• Praha,"LINESTRING(14.450325965881346 50.05789646795757, 14.450948238372803
• 50.05791713211645, 14.45121645927429 50.05803422884852, 14.451420307159424
• 50.058764354966634, 14.451624155044554 50.0591569653608, 14.451828002929688
• 50.059460030836206, 14.45177435874939 50.059522021265785)",http://
• www.opengis.net/def/crs/EPSSG/0/4258
3 442674,Africká,554782,Praha,3100,Hlavní město
• Praha,"LINESTRING(14.349464178085329 50.09616500163075, 14.349786043167112
• 50.096075528282334, 14.349968433380127 50.09575892893866, 14.350258111953734
• 50.09575892893866, 14.350826740264893 50.09566257220141, 14.352350234985352
• 50.09597228959578, 14.354592561721802 50.096399008059755, 14.355171918869019
• 50.096392125534024, 14.355483055114746 50.09633018275811, 14.355762004852295
• 50.09634394782634, 14.358251094818113 50.096791310391346, 14.358723163604736
• 50.096873899946615, 14.359141588211058 50.09697713669046, 14.359956979751587
• 50.09699778401253, 14.36085820198059 50.09715607985288, 14.362660646438599
• 50.09761031718664, 14.363701343536377 50.09788561044452, 14.363905191421507
• 50.09789937506589, 14.364173412322996 50.09786496350506, 14.364463090896606
• 50.09773419934845)",http://www.opengis.net/def/crs/EPSSG/0/4258
4 442682,Akátová,554782,Praha,3100,Hlavní město
• Praha,"LINESTRING(14.41953420639038 50.077619800339754, 14.41979169845581
• 50.077647341221535, 14.421336650848389 50.07749586617591, 14.42479133605957
• 50.077330620125764, 14.429275989532469 50.07717914407949, 14.431657791137694
• 50.07711029117297)",http://www.opengis.net/def/crs/EPSSG/0/4258
```



# **SPATIAL LINKED DATA FORMATS**



## Geo WGS-84



```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:geo="http://www.w3.org/2003/01/geo/wgs84_pos#">
  <geo:Point>
    <geo:lat>49.701</geo:lat>
    <geo:long>14.552</geo:long>
  </geo:Point>
</rdf:RDF>
```



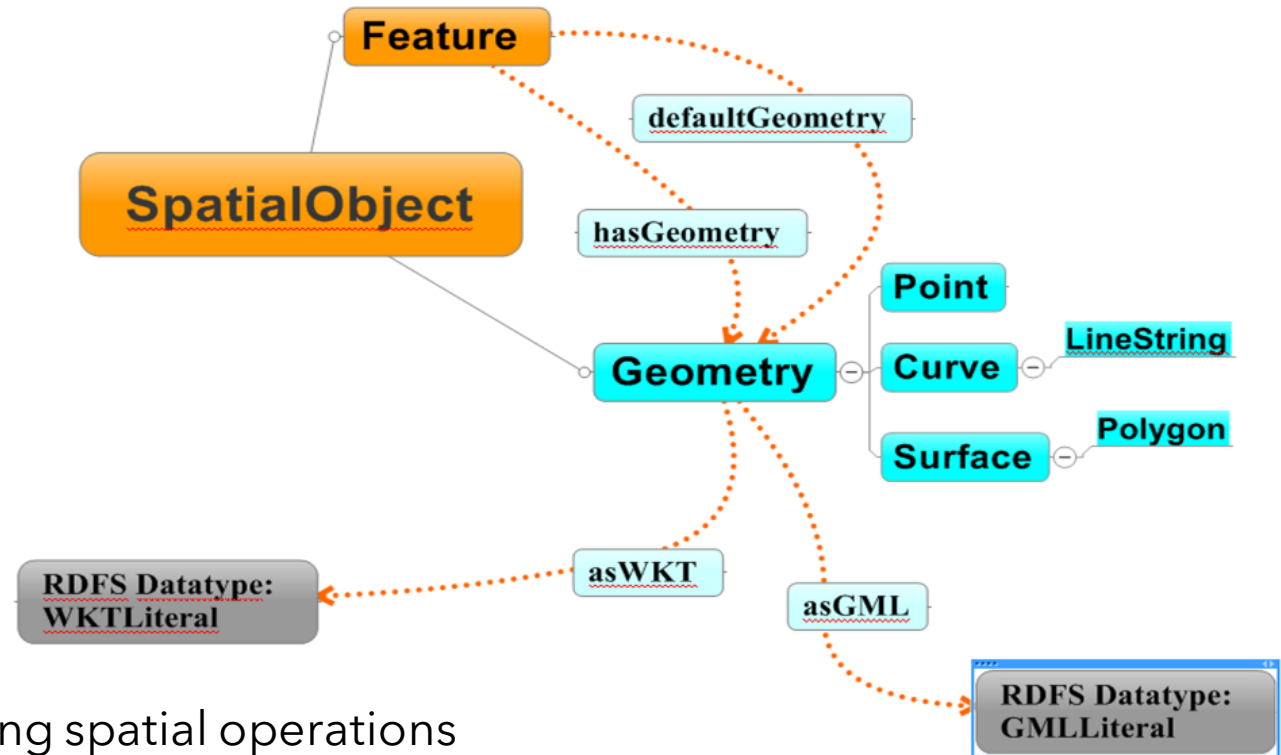
Perfectly represents points in WGS-84



Cannot represent anything else

'lat' and 'long' are standalone tropes (in geo, we usually use 3-letters shorts 'lat' and 'lon')

# GeoSPARQL



Ontology + query language supporting spatial operations  
Geometry representation in both GML and WKT  
Support various CRS



May be too complicated for beginners  
Seems overpowered for simple representation of points



# GeoSPARQL

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix geosparql: <http://www.opengis.net/ont/geosparql#> .
@prefix ds-par: <http://onto.fel.cvut.cz/ontologies/town-plan/parcely/> .
@prefix databaseTableParcely: <http://onto.fel.cvut.cz/ontologies/town-plan/databaseTableParcely/>
@prefix par-geometry: <http://onto.fel.cvut.cz/ontologies/town-plan/parcelakn_dokm_p/geometry/>
@prefix townplan: <http://onto.fel.cvut.cz/ontologies/town-plan/>
```

```
townplan:parcelakn_dokm_p/1/2018-01-29T14:36:24.178617 a ds-par:Parcely,
    geosparql:Feature ;
    rdfs:label "parcelakn_dokm_p/1/2018-01-29T14:36:24.178617" ;
    databaseTableParcely:dat_vznik "2008-09-25"^^xsd:date ;
    databaseTableParcely:existujedi "A" ;
    databaseTableParcely:id 2087553101.0 ;
    databaseTableParcely:id_poskyt 397 ;
    databaseTableParcely:katuze_kod 727164 ;
    databaseTableParcely:nazev_ku "Vinohrady" ;
    databaseTableParcely:ogc_fid 1 ;
    databaseTableParcely:par_id 2087553101.0 ;
    databaseTableParcely:parcela "1057" ;
    databaseTableParcely:shape_area 260.475900002 ;
    databaseTableParcely:shape_length 65.6304823872 ;
    databaseTableParcely:tid_parcelakn_dokm_p 61534.0 ;
    databaseTableParcely:vymera 260 ;
    geosparql:hasGeometry par-geometry:1/2018-01-29T14:36:24.178617 .
```

```
par-geometry:1/2018-01-29T14:36:24.178617 a geosparql:Geometry ;
    rdfs:label "parcelakn_dokm_p/geometry/1/2018-01-29T14:36:24.178617" ;
    geosparql:asWKT "MULTIPOLYGON((( (-742241.02 -1045480.81,-742242.84 -1045482.35,
        -742257.059 -1045469.76,-742246.0798 -1045456.9,-742237.98
        -1045465.82,-742241.02 -1045480.81))))" .
```





# GeoJSON-LD

```
{
  "@context": https://ofn.gov.cz/umístění/2020-07-01/kontexty/umístění.jsonld,
  "typ": "Umístění",
  "název":
  {
    "cs": "Národní park Šumava"
  },
  "geometrie":
  {
    "type": "Point",
    "coordinates": [13.6309462, 48.9720309]
  }
}
```

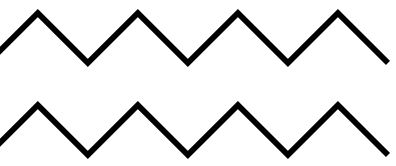
```
{
  "@context":
  {
    "@version": 1.1,
    "locn": "http://www.w3.org/ns/locn#",
    "dcterms": http://purl.org/dc/terms/,
    ...
    "geometrie":
    {
      "@id": "locn:geometry",
      "@context": "https://geojson.org/geojson-ld/geojson-context.jsonld"
    }
  }
}
```





# **S P A T I A L R E L A T I O N S A N D O P E R A T I O N S**



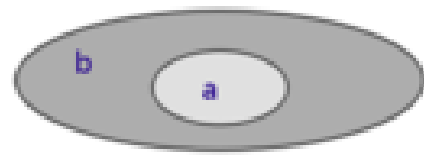


# Spatial relations

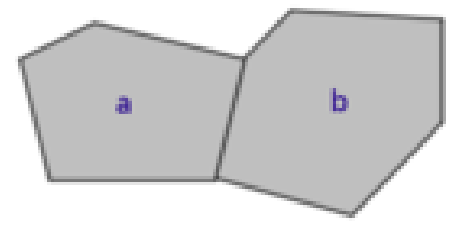
Relation between two (or more) spatial objects, usually based on location and/or shape:

- **Topological**
- Directional
- Distance
- Temporal

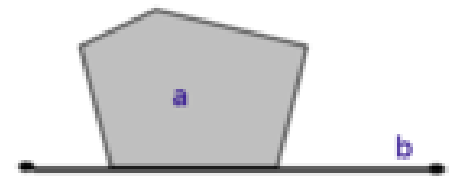
Within(a,b)



Touches(a,b)



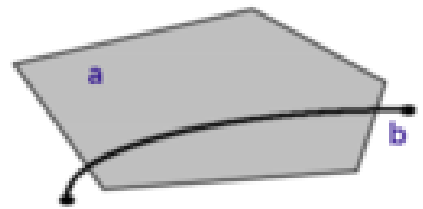
Touches(a,b)



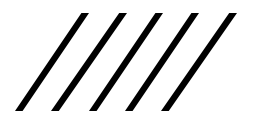
Crosses(a,b)

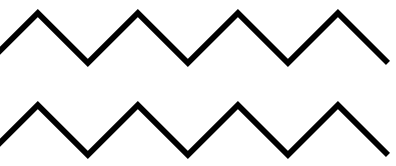


Crosses(a,b)

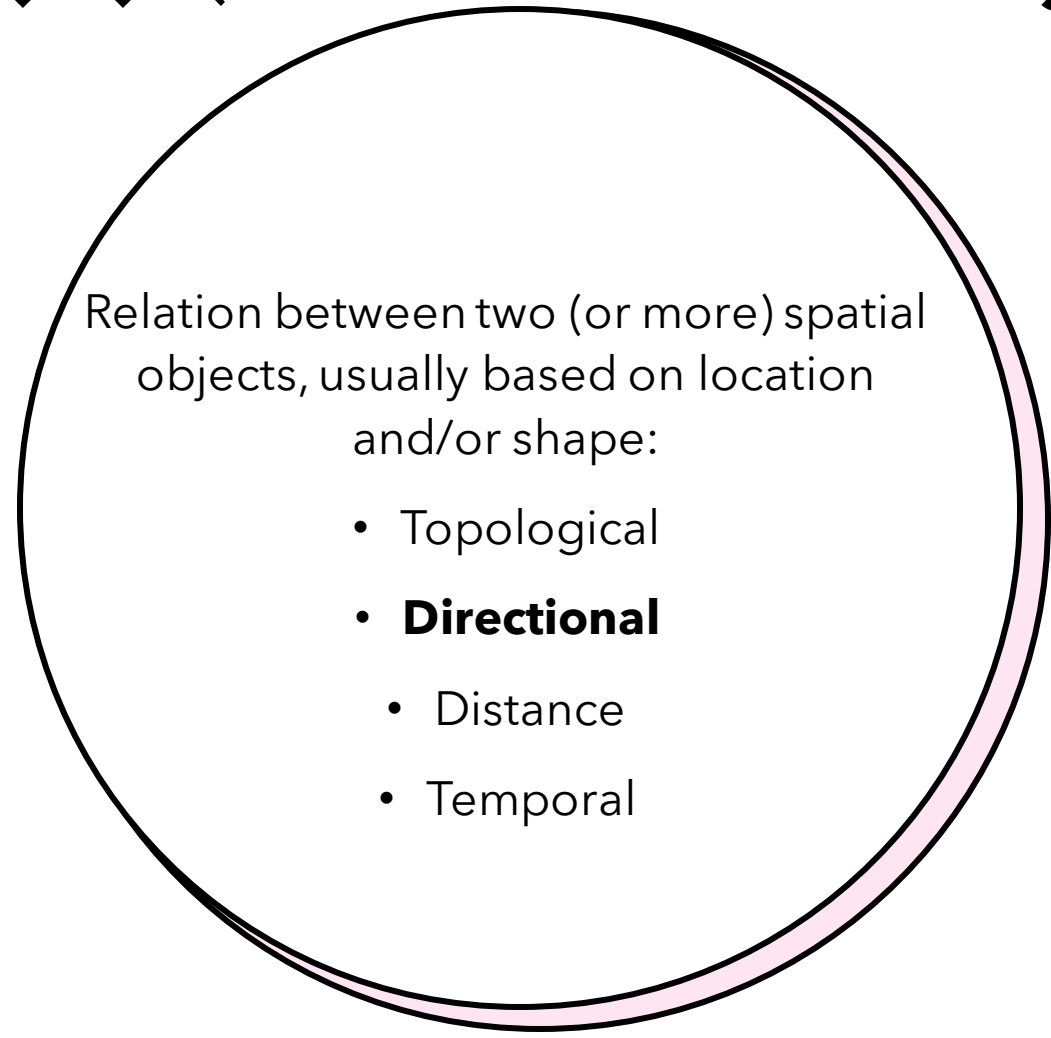


Overlaps(a,b)





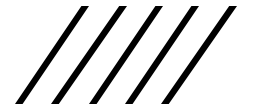
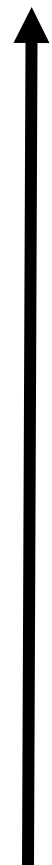
# Spatial relations

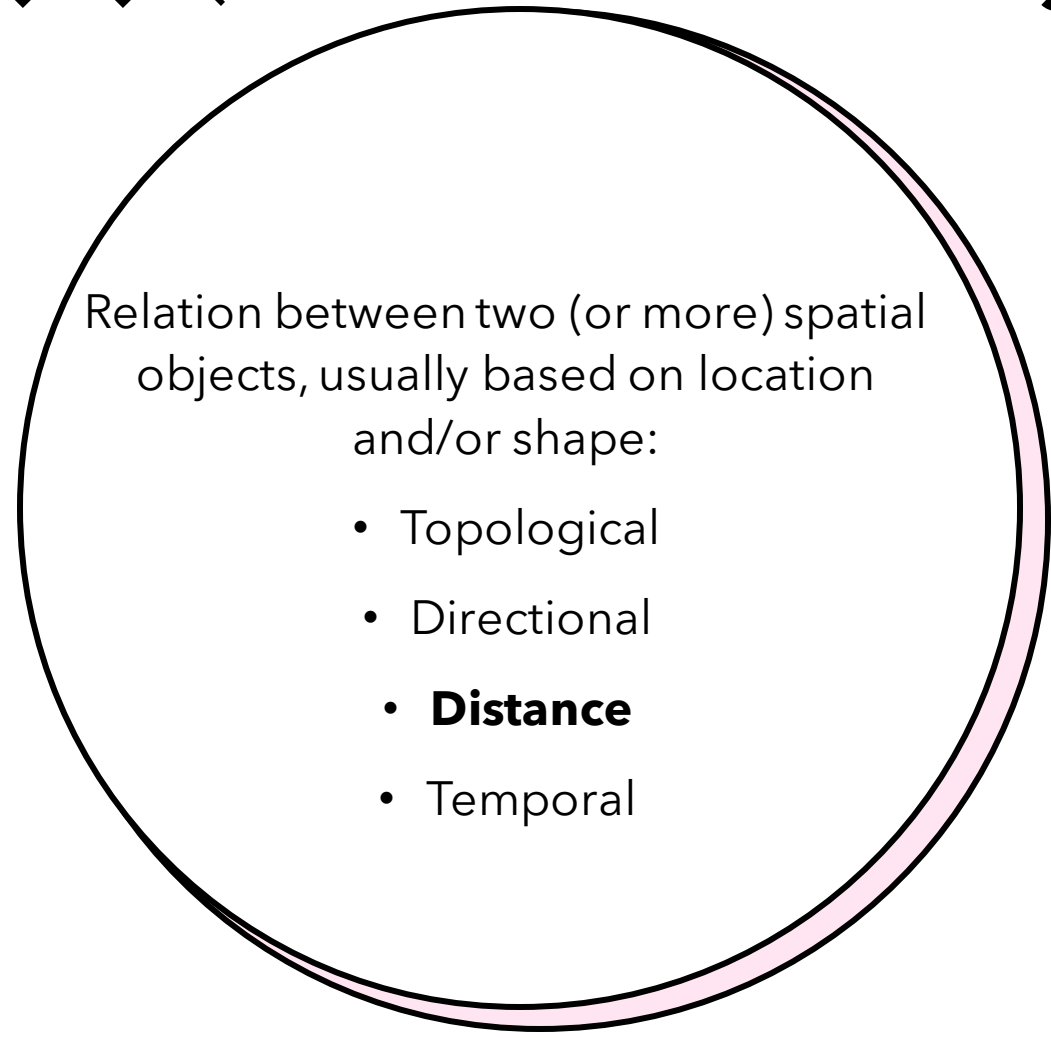
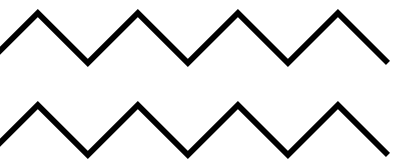


**Left**

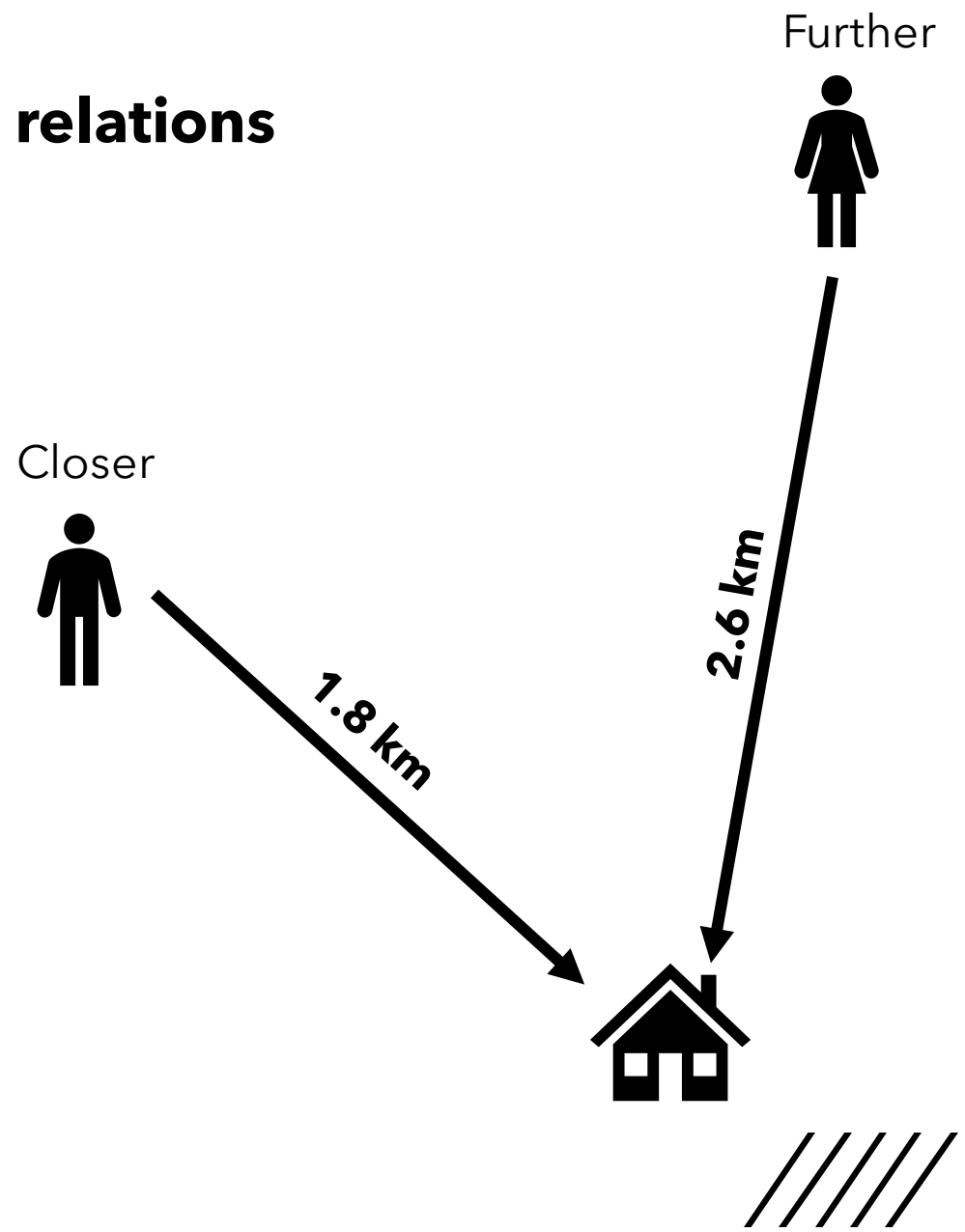


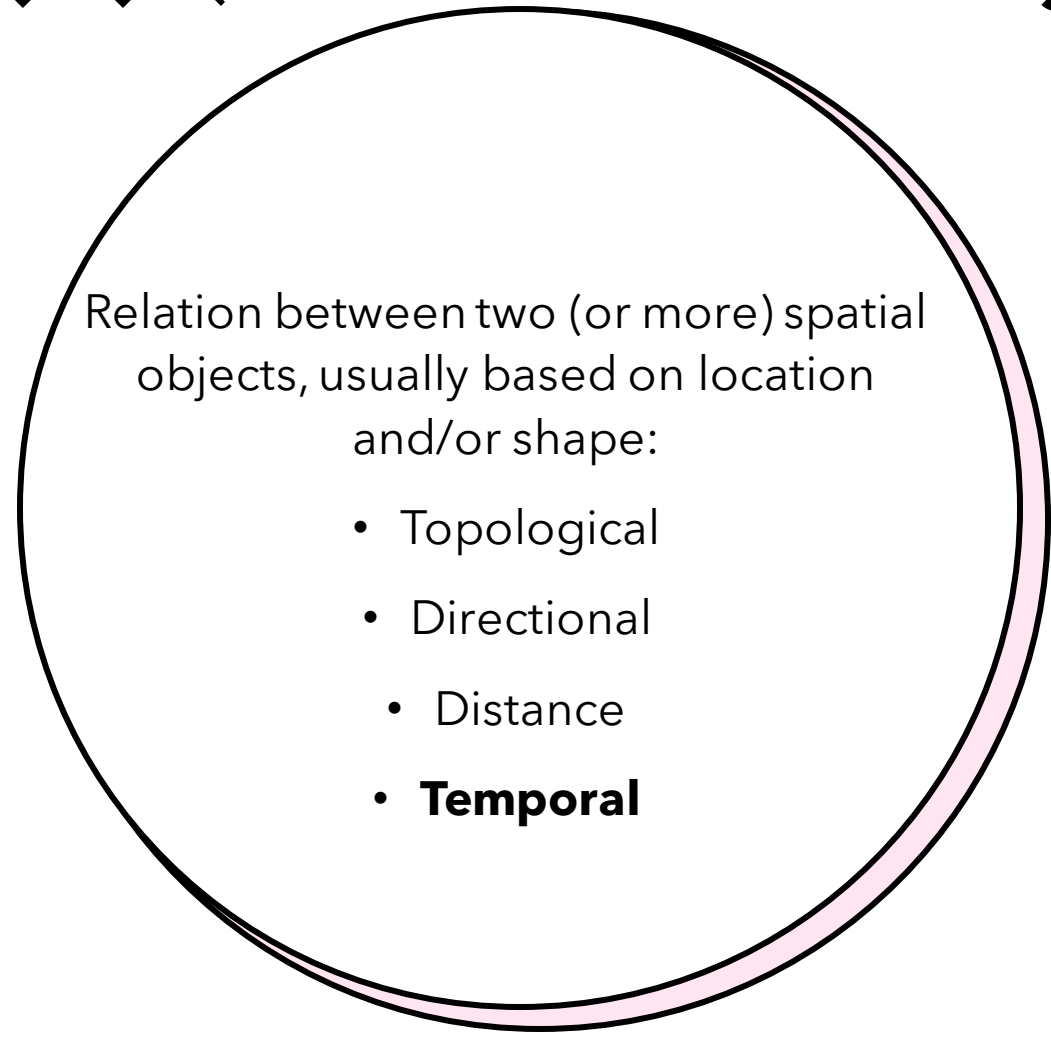
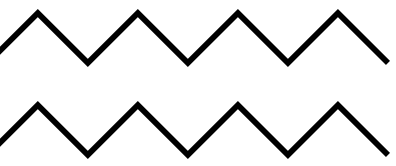
**Right**



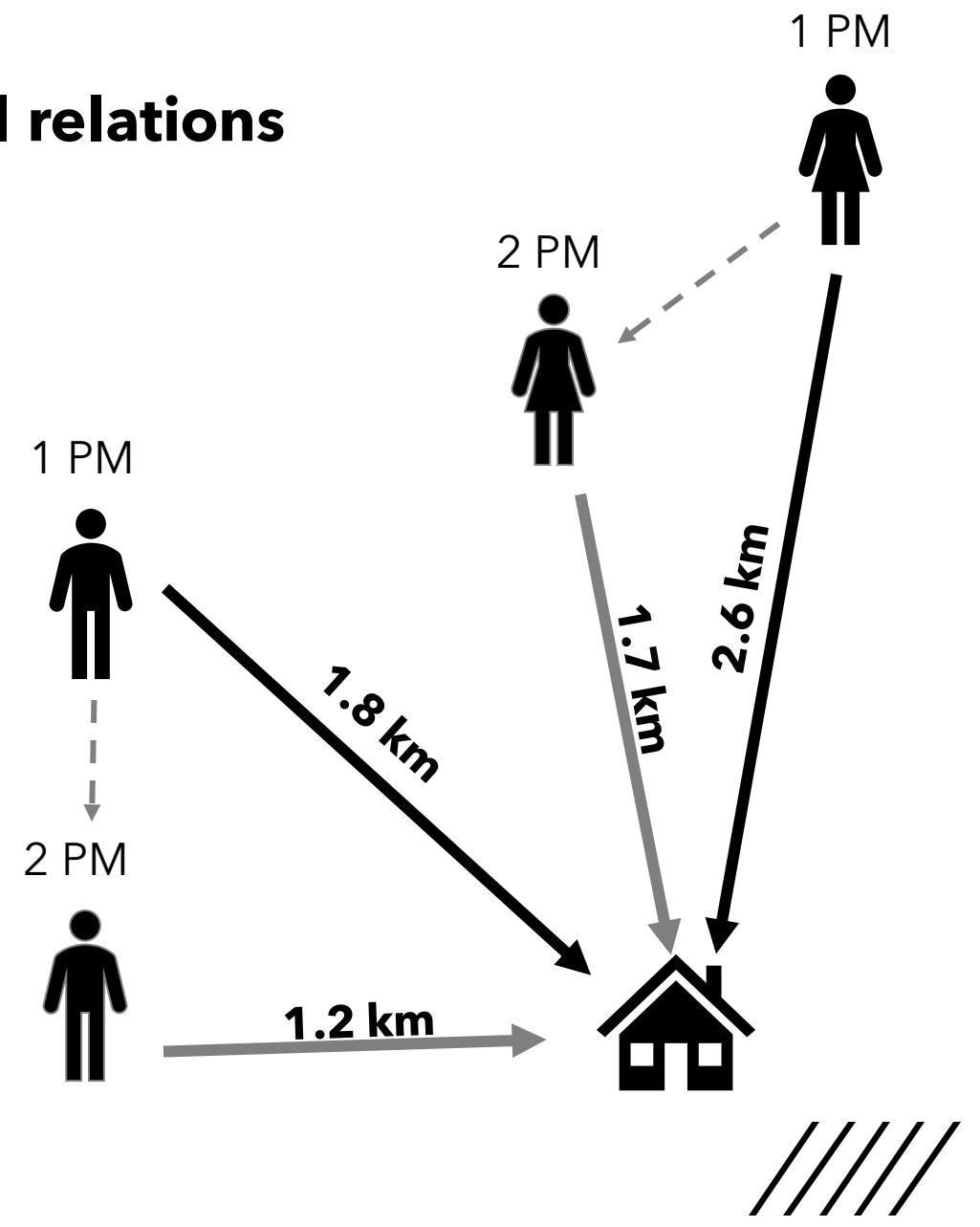


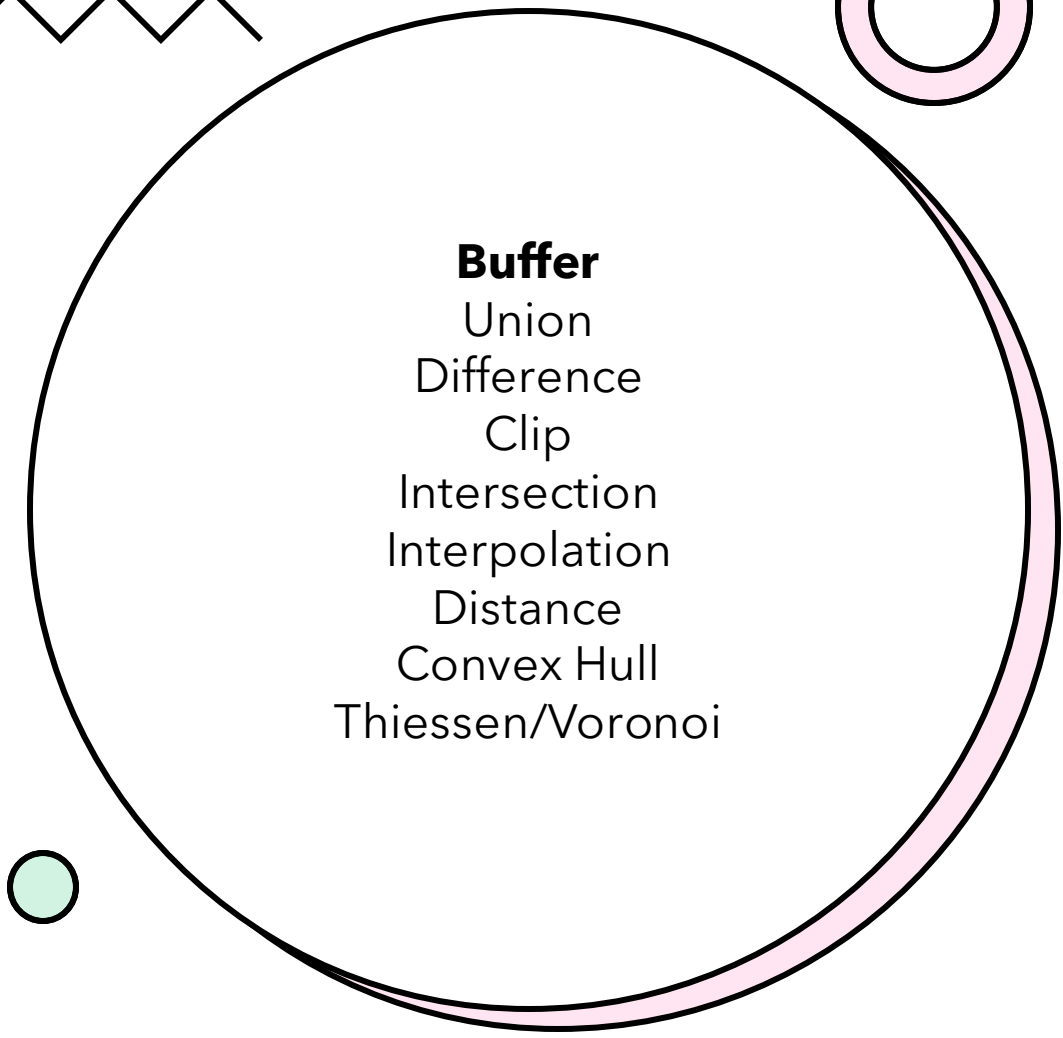
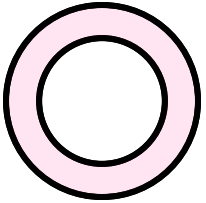
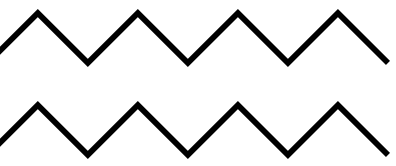
## Spatial relations





## Spatial relations





## Spatial operations



*Buffering a linestring*



*Buffering a polygon with one interior ring*



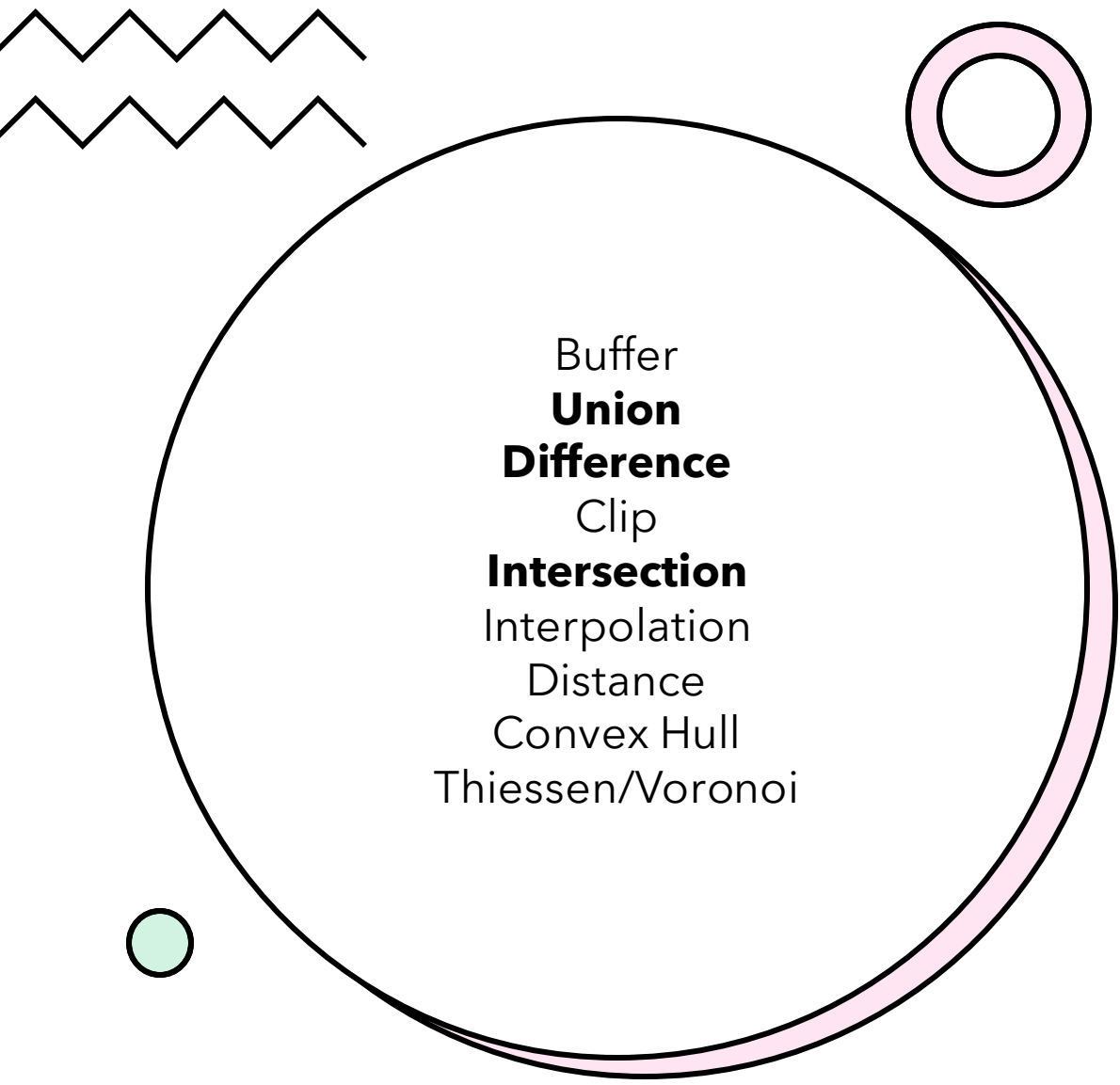
*Buffering a point*



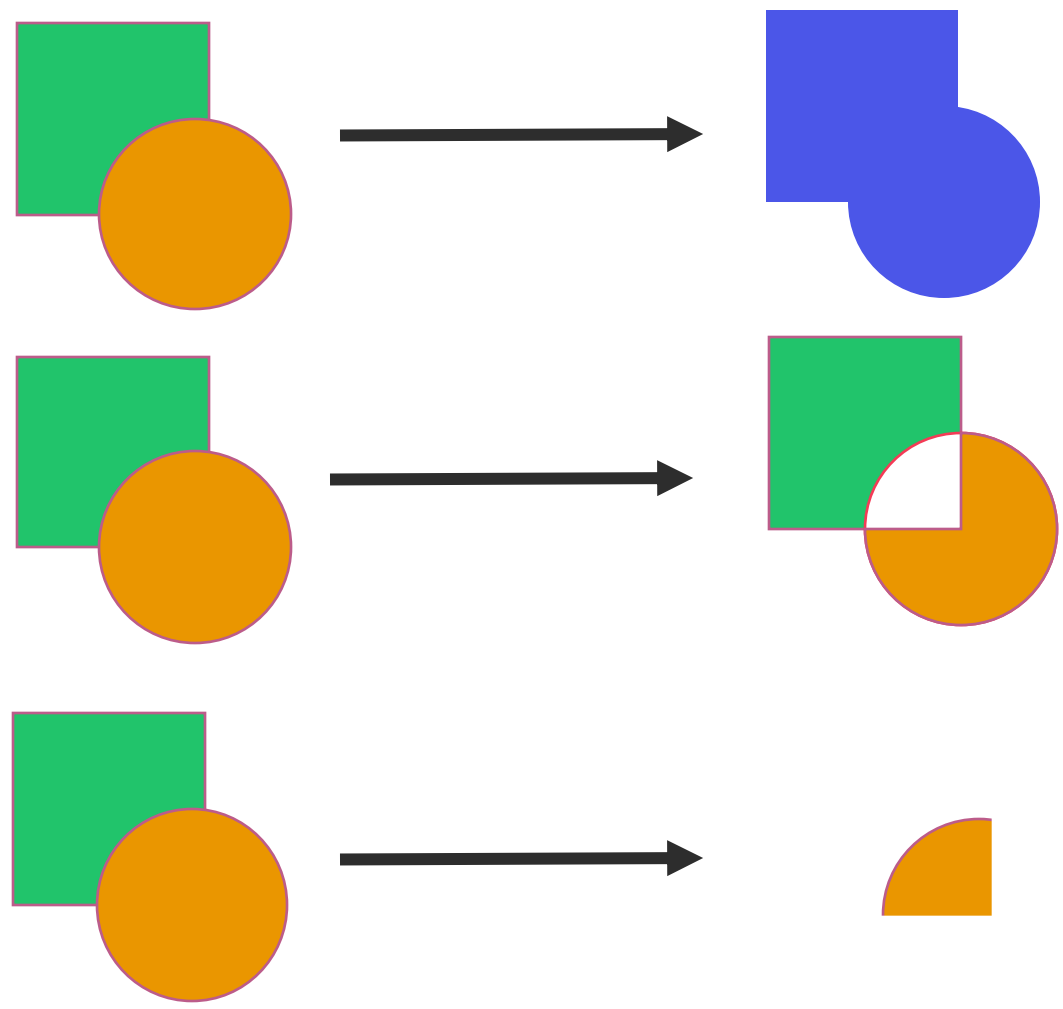
*Buffering a multipoint*

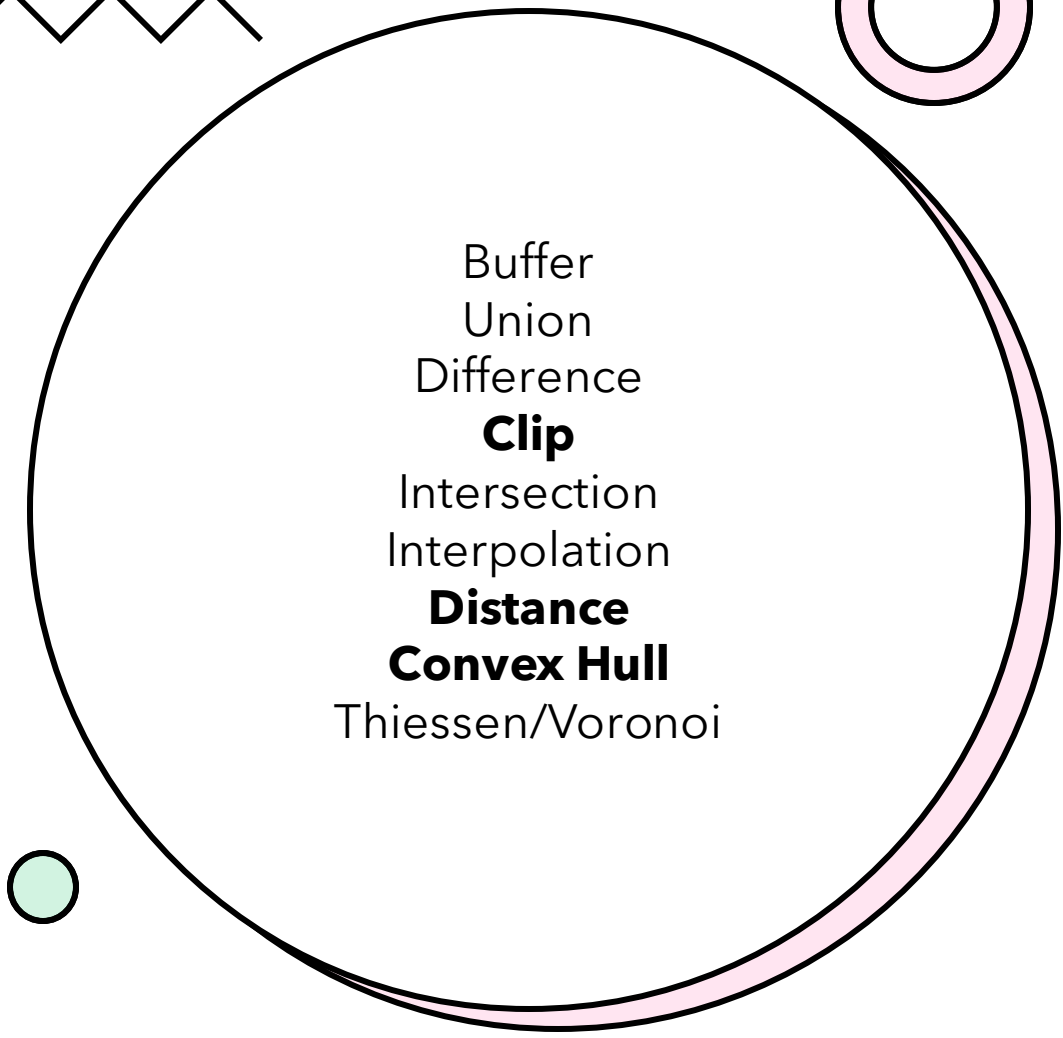
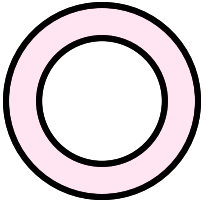
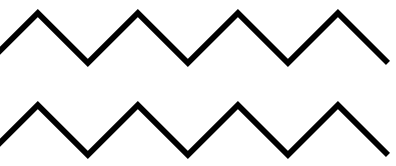




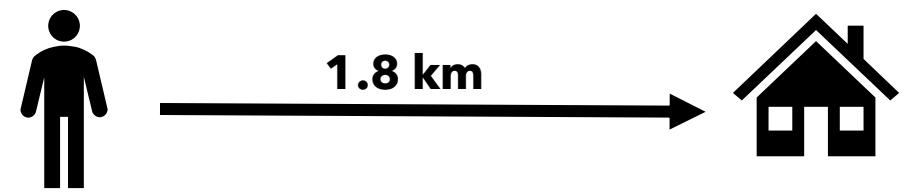


## Spatial operations



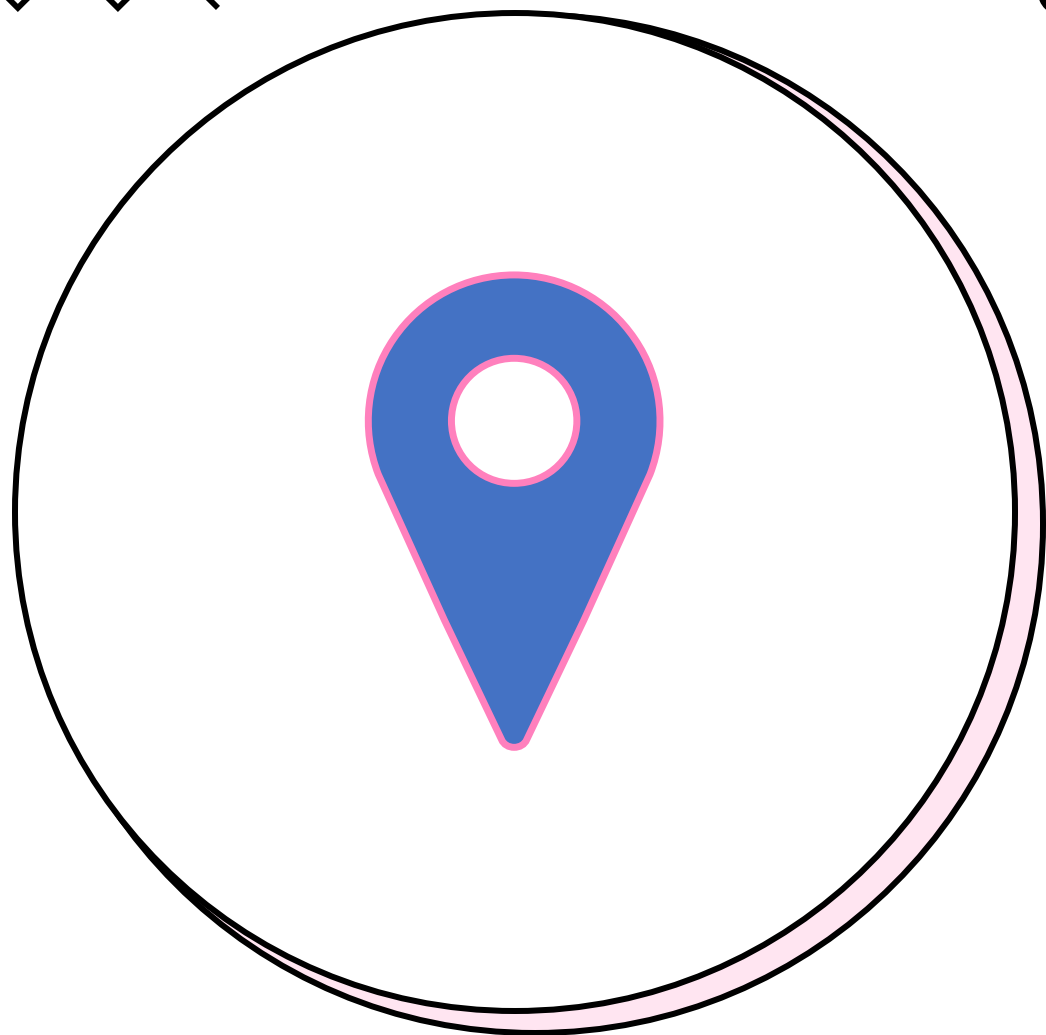


## Spatial operations





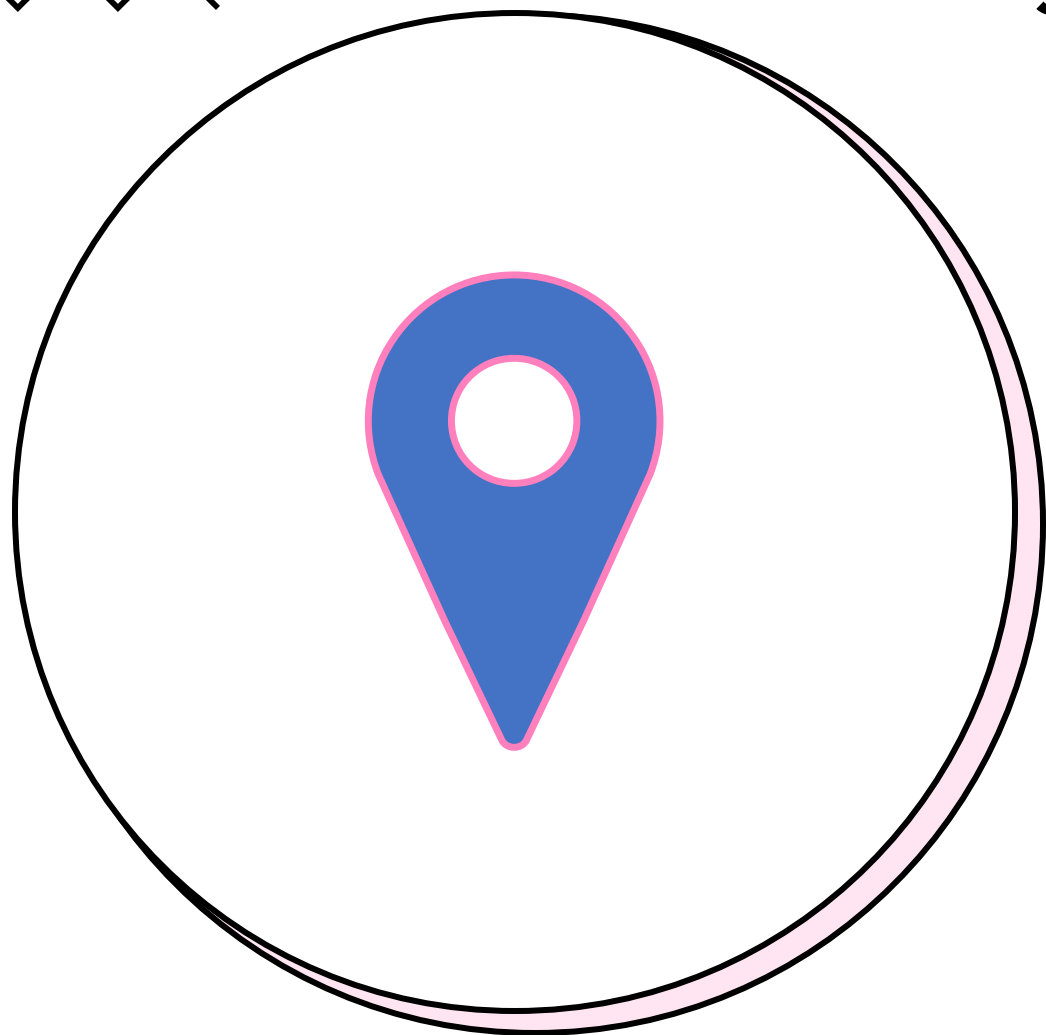
**GIS  
SOFTWARE  
AND  
SPATIAL  
LIBRARIES**



## GIS Software

- QGIS
  - <https://qgis.org/en/site/>
  - Very powerful open source projects
- PostGIS
  - Spatial Extension for PostgreSQL
- ESRI ArcGIS
  - <https://www.arcgis.com/index.html>
  - large commercial project

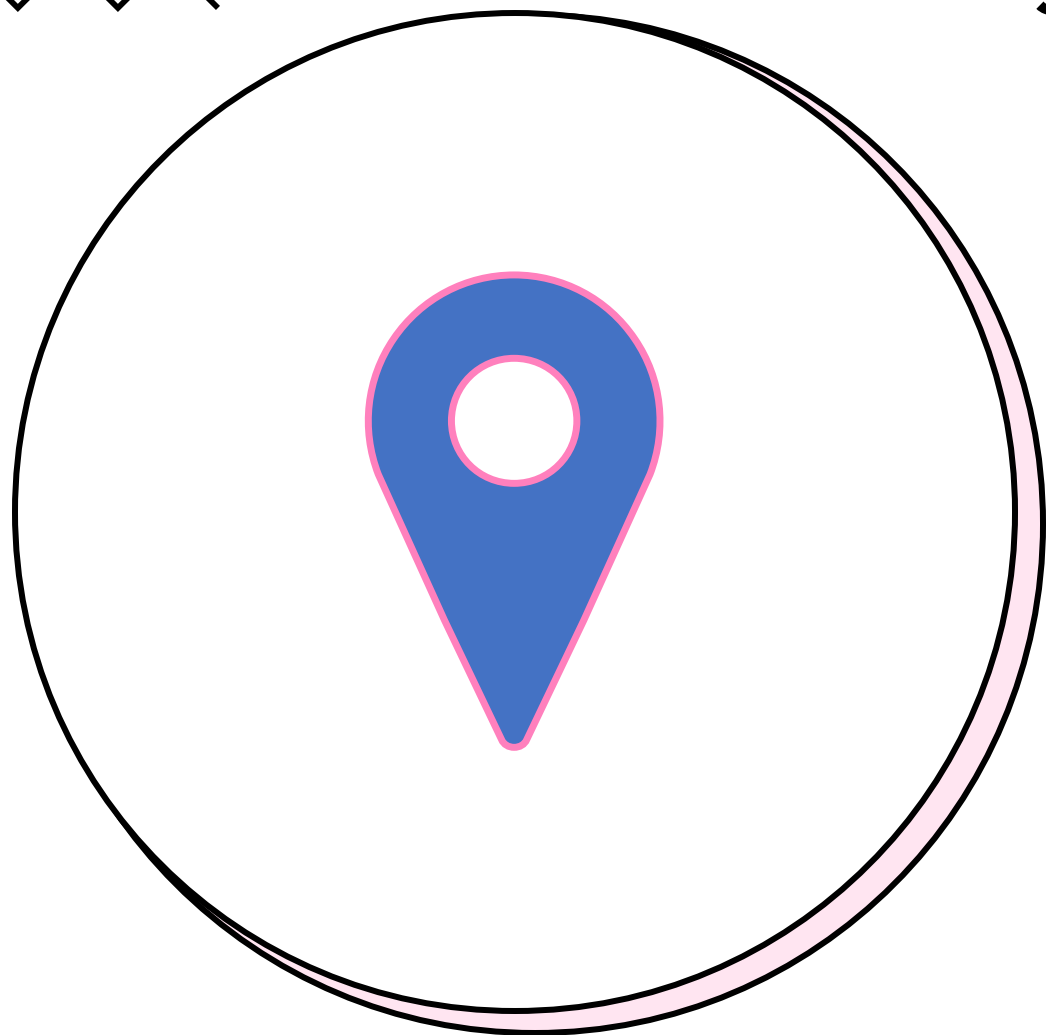




## Spatial libraries

- Leaflet
  - <https://leafletjs.com/>
  - Lightweight JS library for maps
  - Also as a react component
- OpenLayers
  - <https://openlayers.org/>
  - JS API for maps
- MapServer, GeoServer
  - "heavy" solutions
  - Data stored in spatial database on the server, supports wide portfolio of operations
  - Usually used for serving data (as data or maps)





## Spatial libraries

- Geopandas
  - <https://geopandas.org/>
  - Spatial extension for well known python library
  - Also as a react component
- ogr2ogr
  - <https://gdal.org/>
  - Most used library for spatial operations and transformations
  - Used by most listed software
  - Runs from terminal, python, java...

