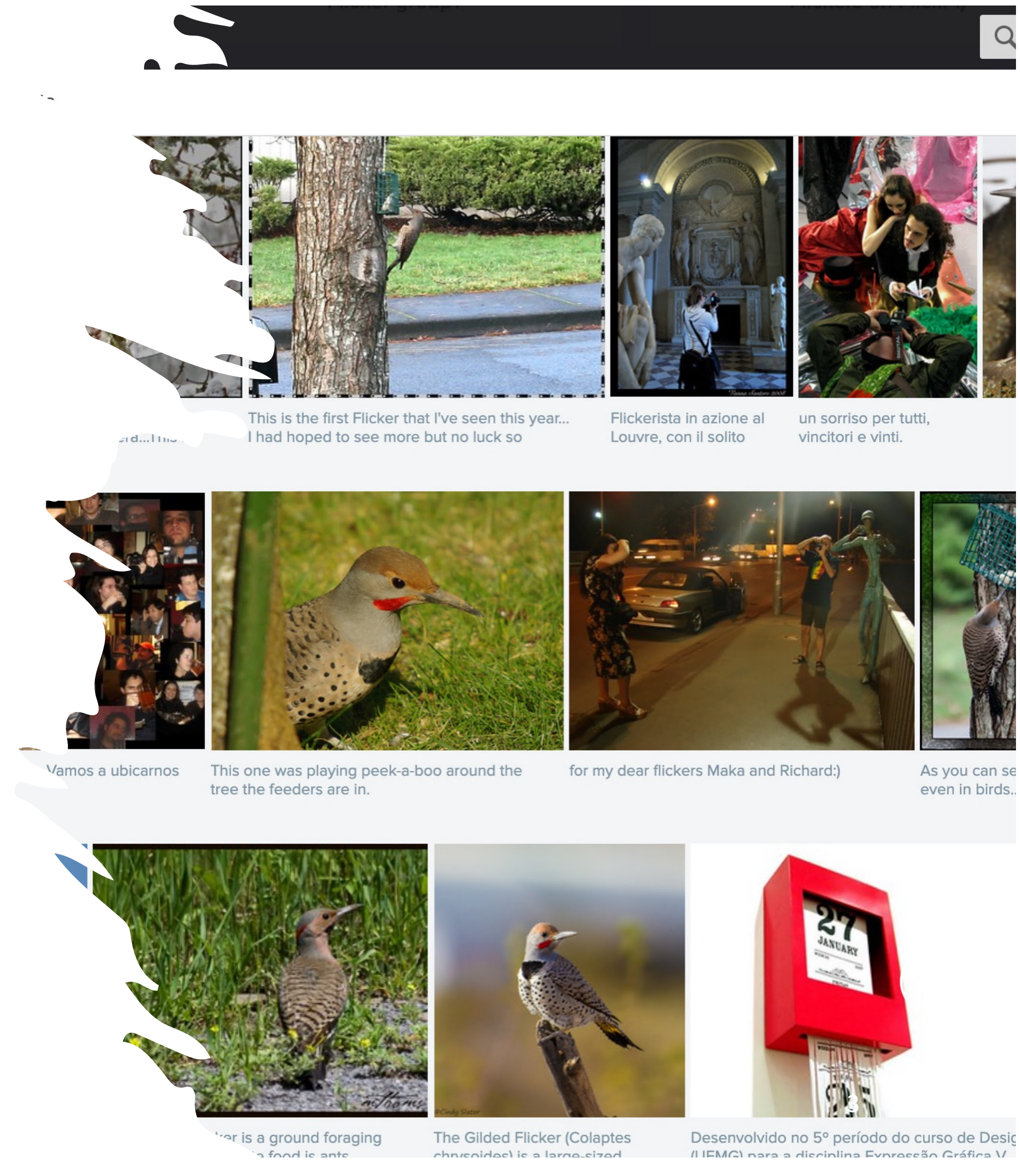
Spatial Data Management using Spatial Databases

Yao-Yi Chiang
Computer Science and Engineering
University of Minnesota
yaoyi@umn.edu

What is Data Management?

How do you manage your photos?

- Most cellphones take nice photos
  - Taking 3 photos a day will give you ~1,000 photos a year
  - Taking a 5-day vacation would give you 200 photos
- Ways to managing photos
  - Leave them on the phone?
  - Organize them into folders?
  - Upload them to some cloud services?
- Which method is the best?



Considerations for Managing Photos

Find photos by time

Find photos by subjects

Find photos by locations

Searching must be fast!

Photos need to be secure

Available resources

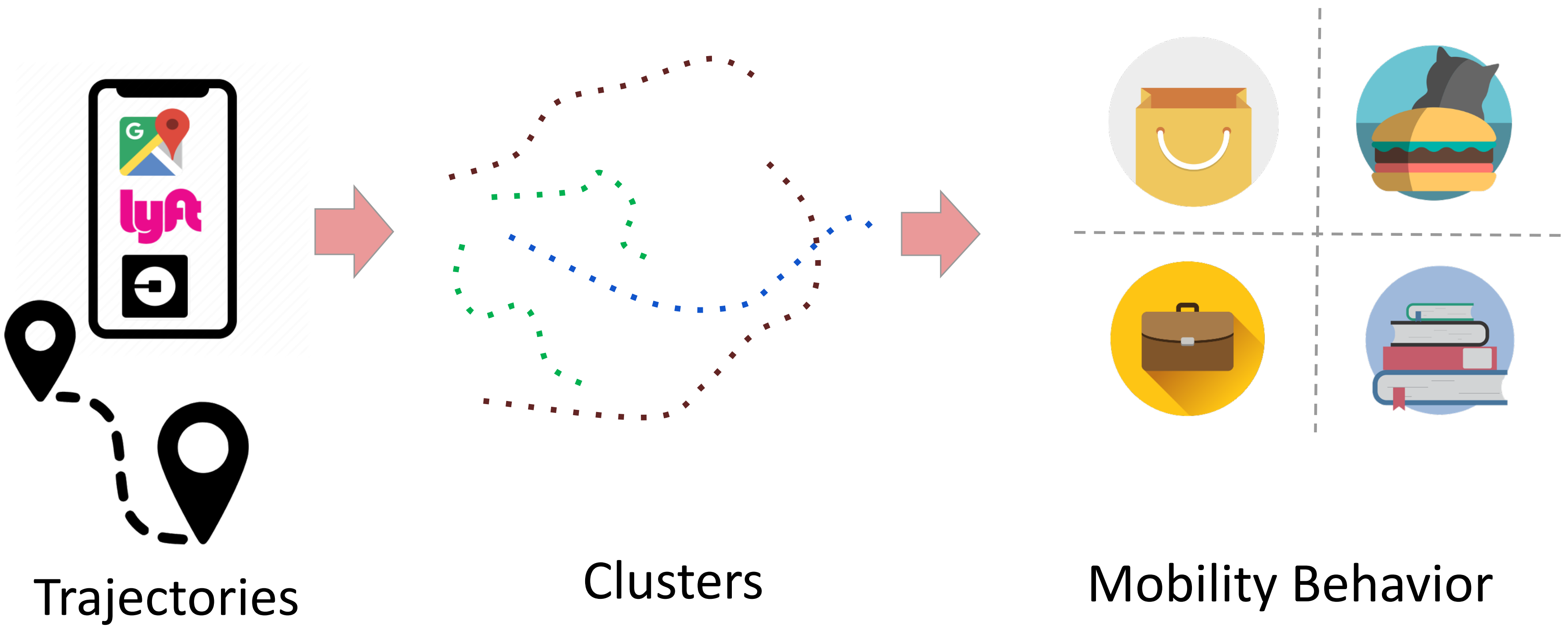
Data Management (Oracle)

- Data management is the practice of collecting, keeping, and using data securely, efficiently, and cost-effectively.
- help people, organizations, and connected things
  - optimize the use of data within the bounds of policy and regulation
  - (use data to) make decisions and take actions that maximize the benefit to the organization

https://www.oracle.com/database/what-is-data-management/

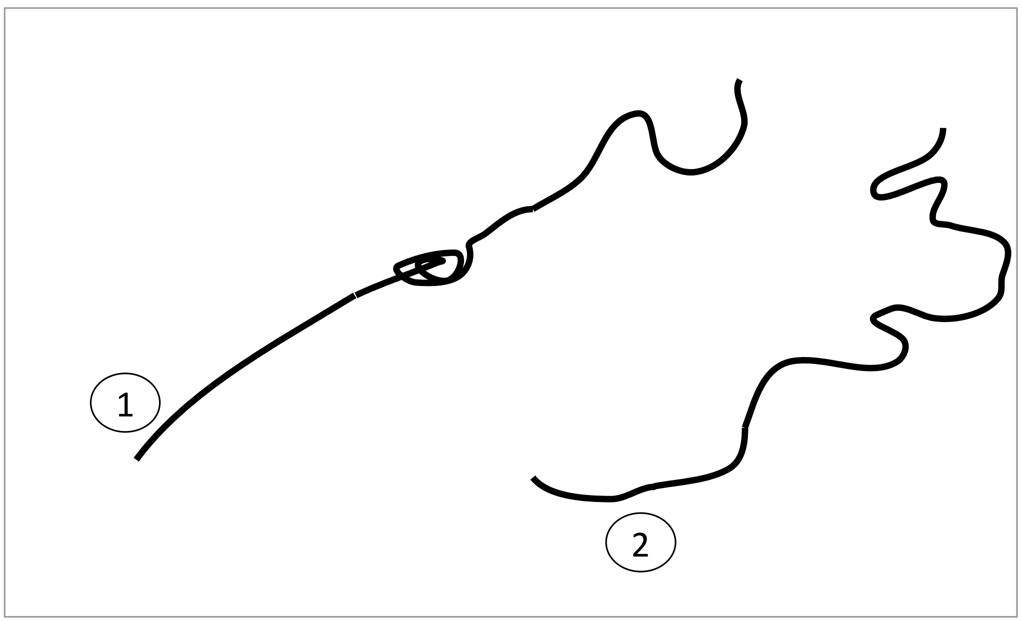
What are Some Data Use Cases for Spatial AI?

Trajectory Mining

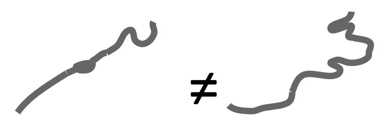


Yue, M., Li, Y., Yang, H., Ahuja, R., **Chiang, Y.-Y.**, and Shahabi, C. (December 2019). DETECT: Deep Trajectory Clustering for Mobility-Behavior Analysis. In *Proceedings of the 2019 IEEE International Conference on Big Data (Big Data)*, pp. 988–997, Los Angeles, CA, USA

Do these two trajectories have the same moving behavior?

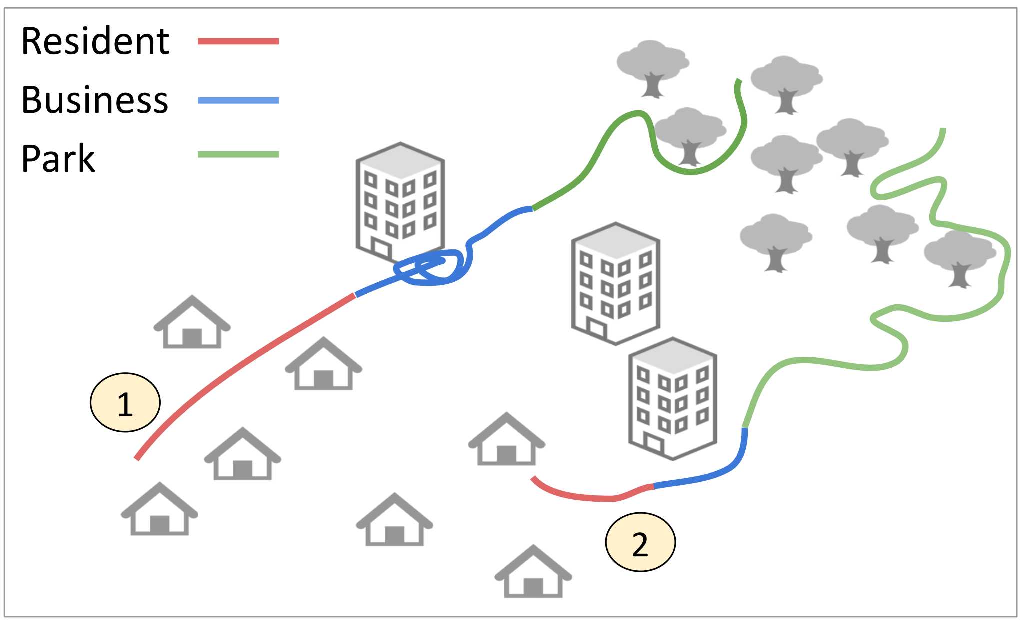


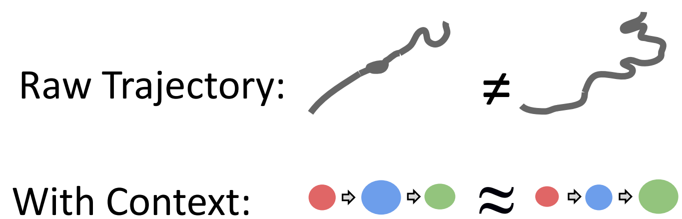
Shapes ✖ Distance ✖



With Geographical Context

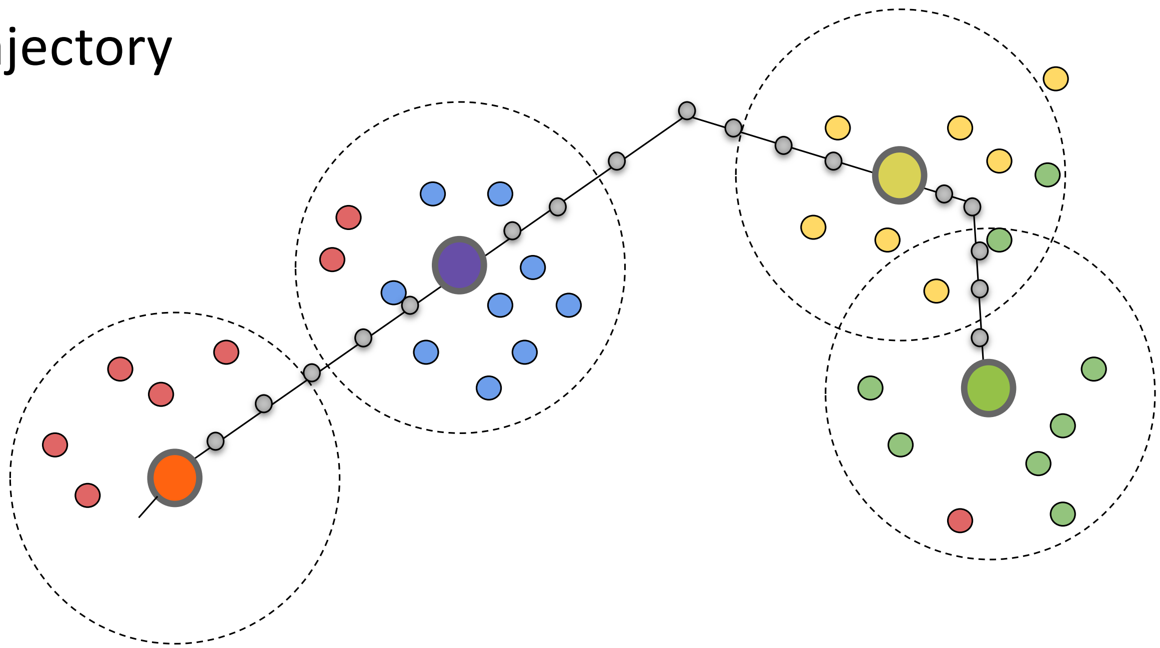
Shapes ✗ Distance ✗ Sequence of activities ✓





Generate Geographic Context

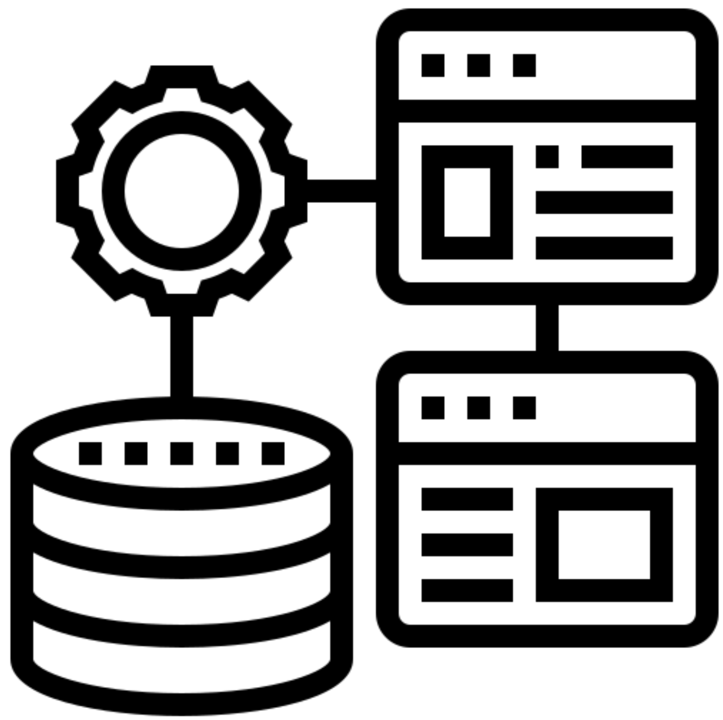
- Find nearby geographic features at each important location of a trajectory
- Distance query



Database Management System

Database Management System (DBMS)

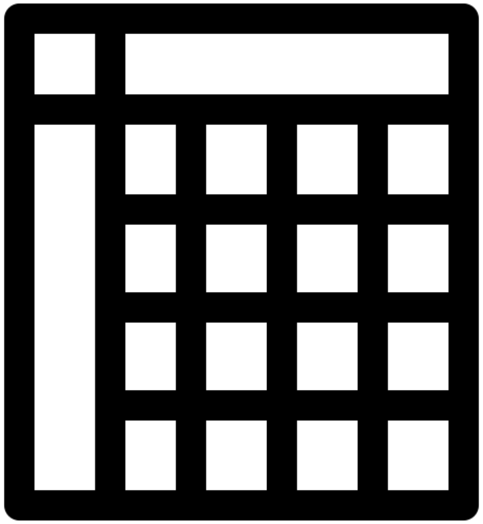
- Persistence across failures
  - maintain a consistent system after failures – software, hardware, network failures, etc.
- Concurrent access to data
- Scalability to search on very large datasets (which do not fit inside main memories of computers)
- Efficiency



https://www.flaticon.com/authors/eucalyp

Tables in Relational DBMS

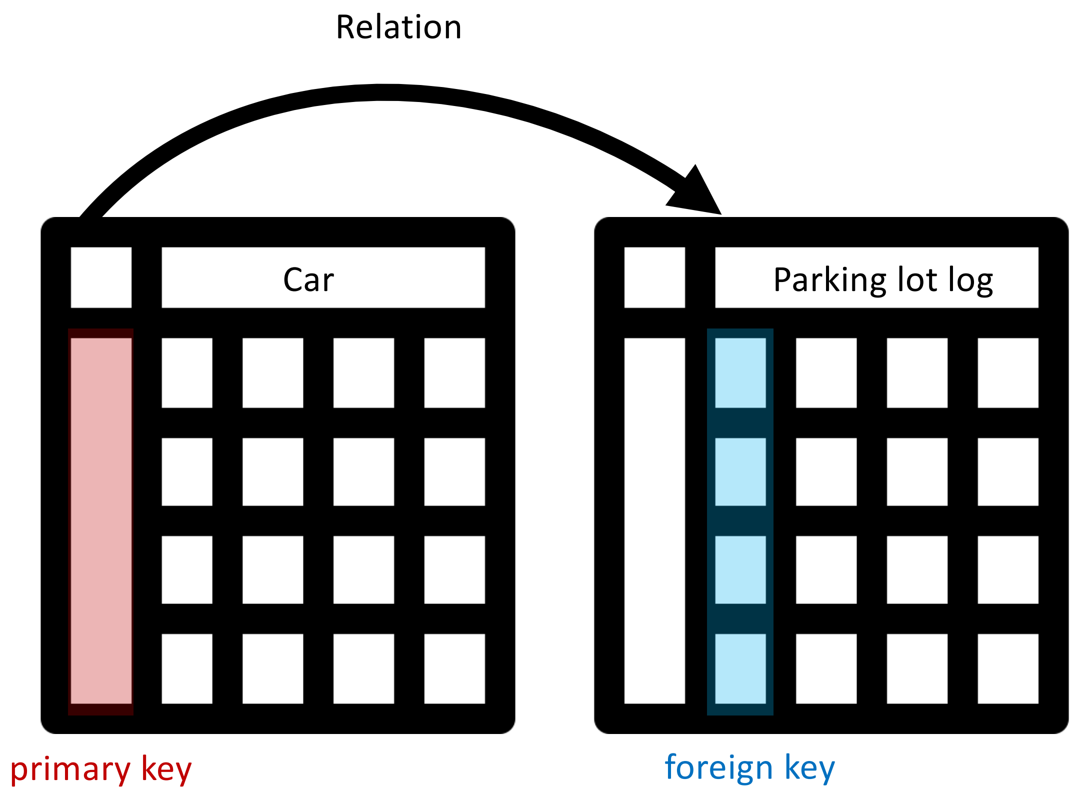
- A table describes the abstraction of a collection of things
  - e.g., cars, parking lot logs
- A table can contain many columns and many rows
- The table schema describes the metadata (columns) of these things (rows)
  - e.g., exist time
- Values in one column share the same abstract data types
  - e.g., integer, floating points



Spreadsheet icons created by Freepik - Flaticon

IDs and Relations

- Typically, one column, the primary key, contains the unique ID (identifier) of the described things
  - e.g., student IDs, social security numbers
- This primary key column can exist in other tables to establish a relation
  - the foreign key



Common Abstract Data Types (ABT)

- Integers
  - 1, 5, -2, 9,…
- Floating points
  - 0.12, -1.33, 2.459,…
- Datetime
  - 11/27/1988 1:11 PM PDT
- String/Multi-characters
  - “Spatial AI is fun!”
- Customized Domains
  - “Vehicle”, “Car”, “Truck”

Parking lot log

ID	Vehicle Type	Exit Time	Parking Fee
0	Car	9 am today	2
1	Truck	1/9/2021 3:00 PM CDT	2.5
2	Minivan	Null	3
3	Wagon	1/9/2021 3:00 PM CDT	1.5
3	Car	1/3/2021 2:00 PM CDT	
4	Small car	1/14/2021 5:15 PM CDT	Did not pay

Maintain Data Integrity

Car

ID	Vehicle Type	Owner
0	Car	David
1	Car	Leilei
2	Car	Hui
3	Truck	Mary
4	Truck	Peter
5	Car	Raji

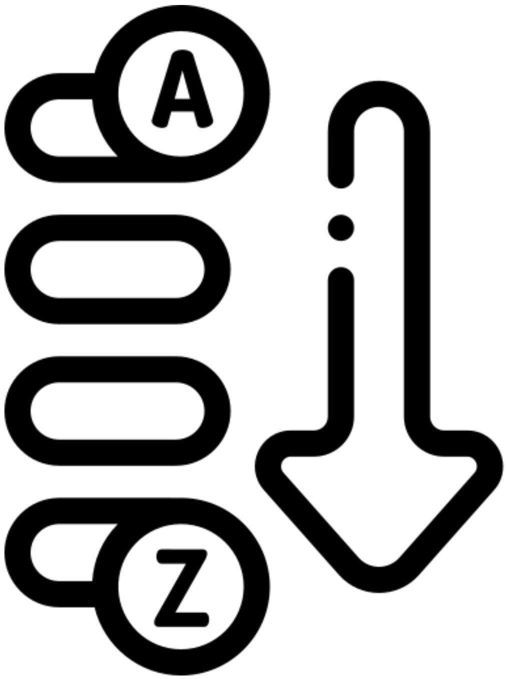
Parking lot log

ID	Vehicle ID	Exit Time	Parking Fee
0	2	9 am today	2
1	3	1/9/2021 3:00 PM CDT	2.5
2	1	Null	3
3	2	1/9/2021 3:00 PM CDT	1.5
3	5	1/3/2021 2:00 PM CDT	
4	1	1/14/2021 5:15 PM CDT	0

Also, take a look at database normalization on Wikipedia: https://en.wikipedia.org/wiki/Database_normalization

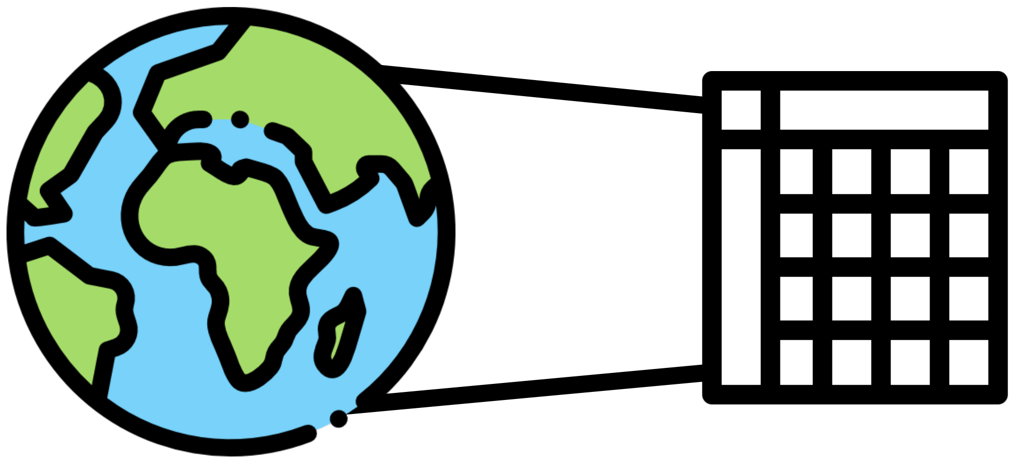
Index

- Take (machine) time to build
- Once built, help speed up data retrieval
- Example:
  - Say you have collected paper questionnaires from 100 randomly selected students
  - Now you need to find information for a specific student using their student ID
  - How do you speed up the process?



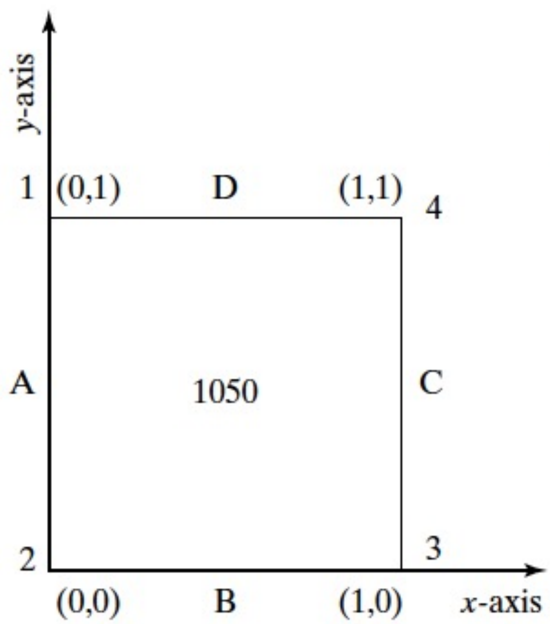
Sort icons created by Freepik - Flaticon

Spatial Data Management System



Model Spatial Data in Traditional DBMS

Use common ADTs, e.g., integer, string, floating points



Census_blocks

Name	Area	Population	boundary-ID
340	1	1839	1050

Polygon

boundary-ID	edge-name
1050	A
1050	B
1050	C
1050	D

Edge

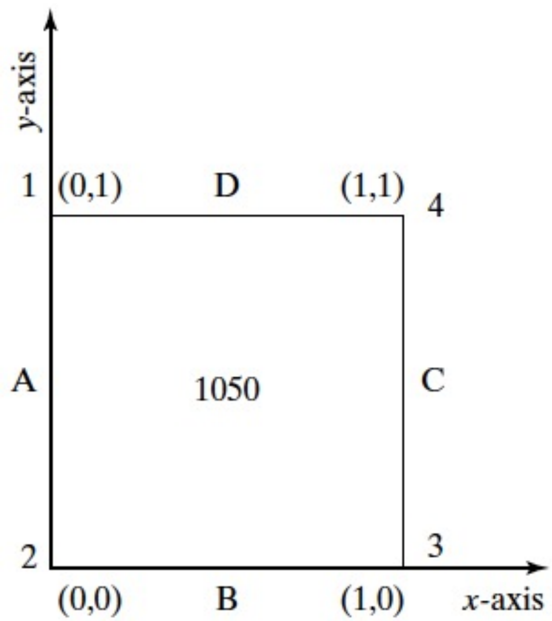
edge-name	endpoint
A	1
A	2
B	2
B	3
C	3
C	4
D	4
D	1

Point

endpoint	*x*-coor	*y*-coor
1	0	1
2	0	0
3	1	0
4	1	1

Model Spatial Data in Traditional DBMS

How about this? What is the data type of the boundary column? String?

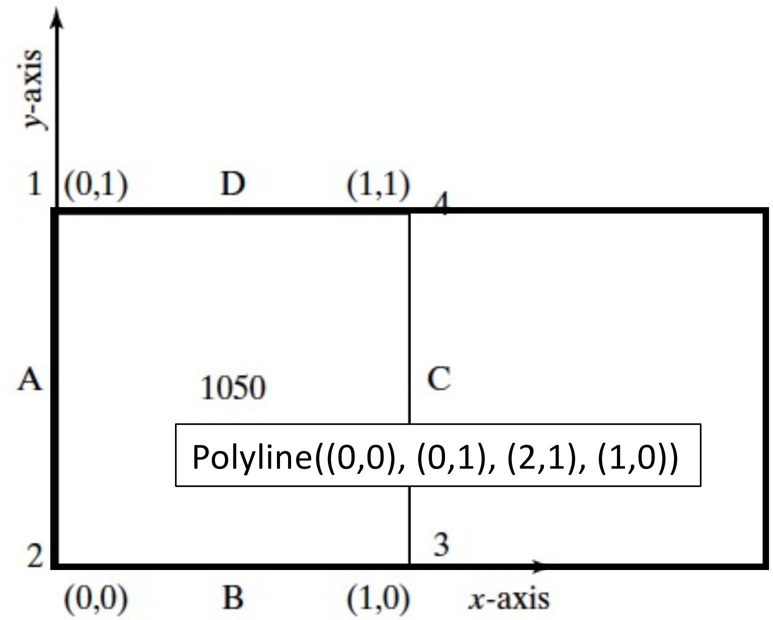


Census_blocks

Name	Area	Population	Boundary
1050	1	1839	Polyline((0,0),(0,1),(1,1),(1,0))

Model Spatial Data in Traditional DBMS

- Can we do
  - Concat(Polyline((0,0), (0,1), (1,1), (1,0)), Polyline(1,0), (2,0), (2,1), (1,1)))
  - And get Polyline((0,0), (0,1), (2,1), (1,0))
- Or ask questions like
  - Area_Size(Polyline((0,0), (0,1), (1,1), (1,0))) = ?
  - And get 1
- We can't if the data type is String.

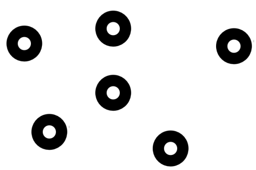


Spatial Data Models

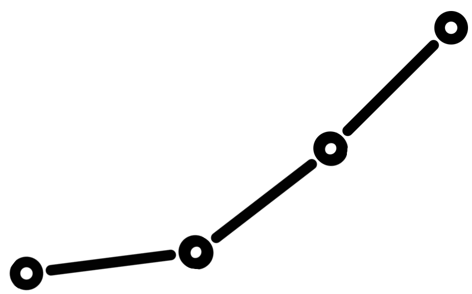
- Spatial data models extends DBMS data models (i.e., requiring a DBMS engine)
- Provide rules to identify identifiable objects and properties of space
- Provide spatial data types at the atomic level, such as integer, string

Spatial Data Vector Model

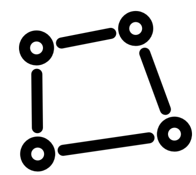
- **Vector** model: manage identifiable things with clear boundaries (or discrete locations), e.g., hotels, mountain peaks, cities, land-parcels



Point



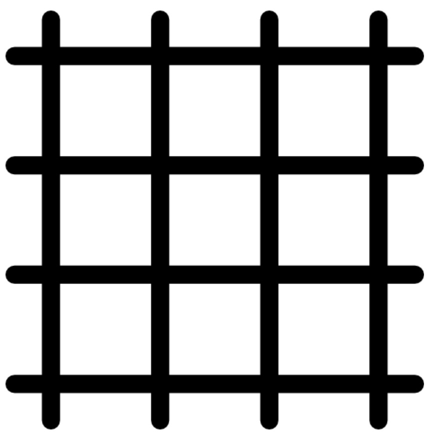
Line



Polygon

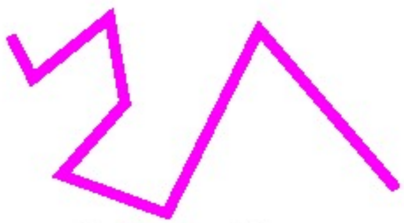
Spatial Data Raster Model

- **Raster** model: manage continuous and amorphous phenomenon, e.g., wetlands, satellite imagery, snowfall

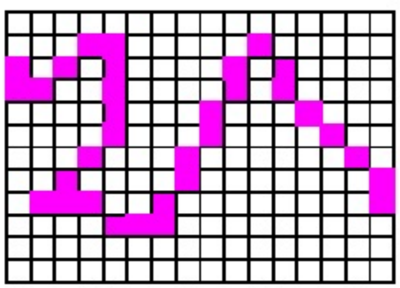


Polygonal icons created by Voysla - Flaticon

Raster and Vector Examples



Vector-based line



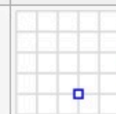
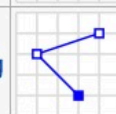
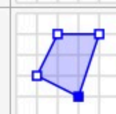
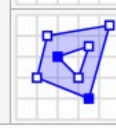
Raster-based line

```
4753456 623412
4753436 623424
4753462 623478
4753432 623482
4753405 623429
4753401 623508
4753462 623555
4753398 623634
```

Flat file

```
0000000000000000
0001100000100000
1010100001010000
1100100001010000
0000100010001000
0000100010000100
0001000100000010
0010000100000001
0111001000000001
0000111000000000
0000000000000000
0000000000000
```

Geometry primitives (2D)

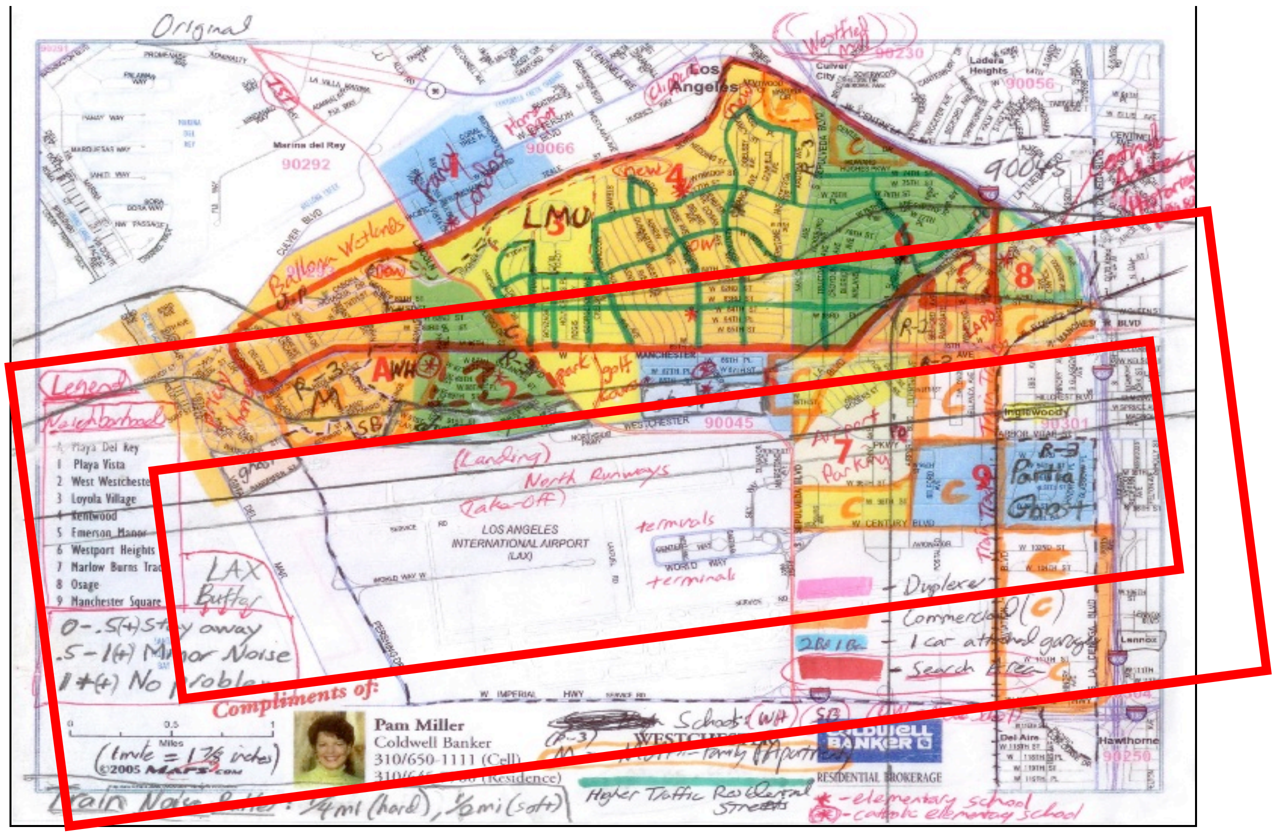
Type	Examples	
Point		POINT (30 10)
LineString		LINESTRING (30 10, 10 30, 40 40)
Polygon		POLYGON ((30 10, 40 40, 20 40, 10 20, 30 10))
		POLYGON ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))

Well-known text (WKT)

Support Common Spatial Data Operations

- Direction queries
  - What is the nearest restaurant north of my current location?
- Distance queries
  - What are the restaurants within a 1-mile radius
- Topology queries
  - How many restaurants within the Los Angeles city boundary?

Spatial Query in Practice



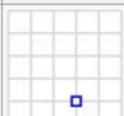
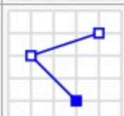
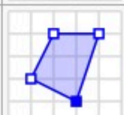
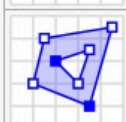
Credit: David Seidner

Spatial Coordinates and Reference Systems

How do we talk about locations on Earth?

- What are these numbers?

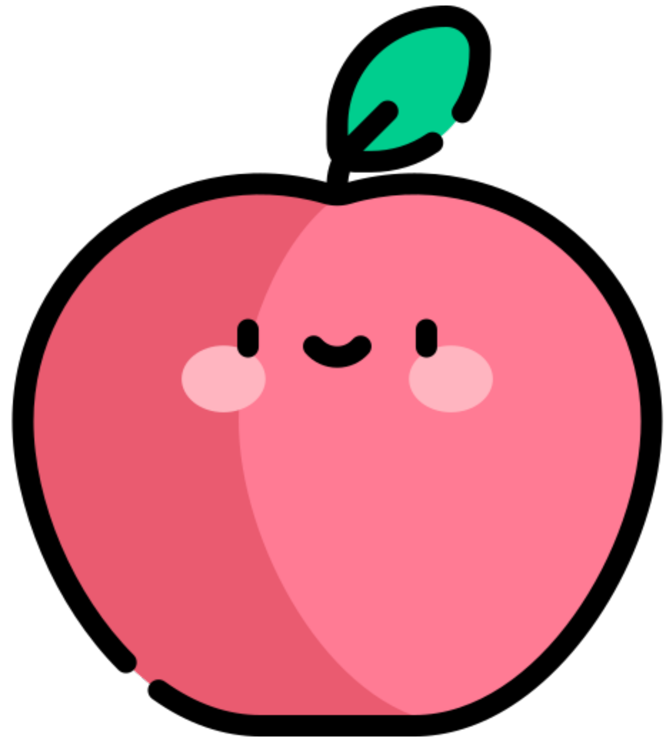
Geometry primitives (2D)

Type	Examples	
Point		POINT (30 10)
LineString		LINESTRING (30 10, 10 30, 40 40)
Polygon		POLYGON ((30 10, 40 40, 20 40, 10 20, 30 10))
		POLYGON ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))

How do we talk about locations on an apple?

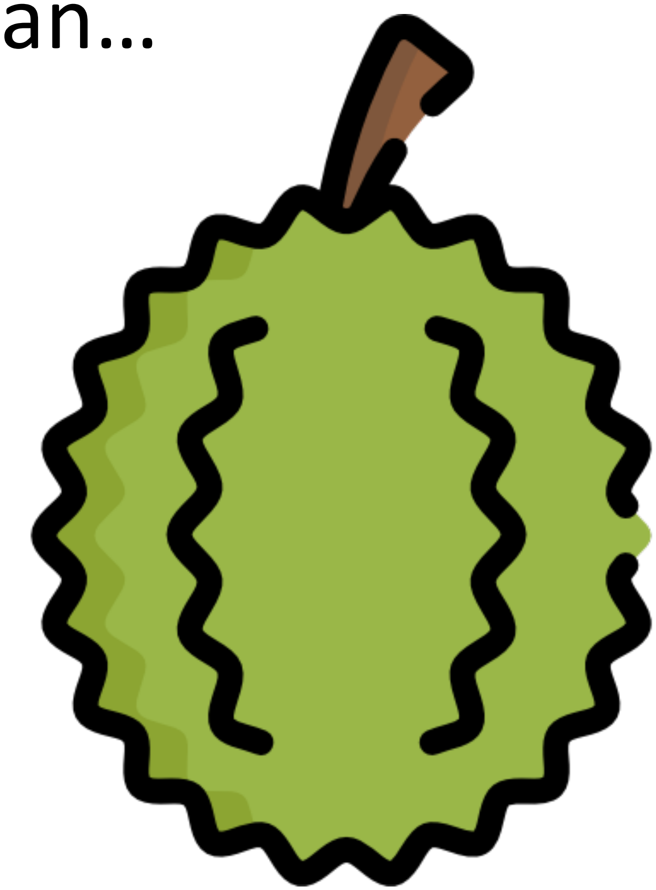
- Label each location on the apple an ID
  - e.g., location 1, location 2, location 3
- Give the apple to other people and use the labels to talk about locations

  OR
- Pick an origin point (e.g., top of the apple)
- Sequentially give each location a number based on its direction and distance to the origin point



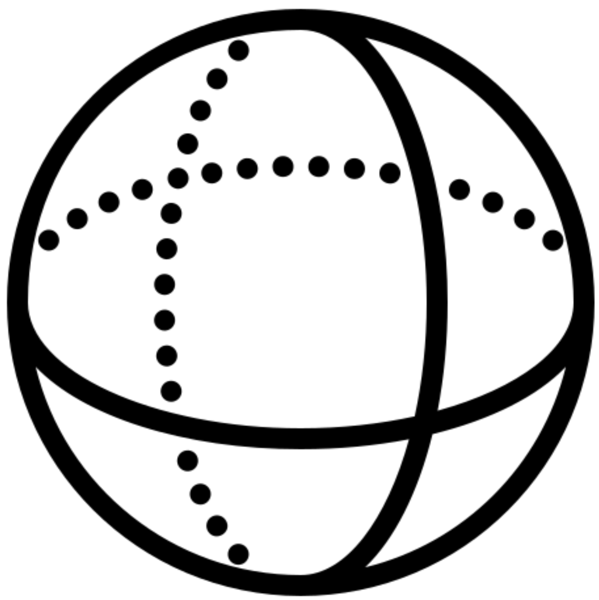
What if it's a durian…

- Also, we don't want to have to pass the durian to other people
- Build a 3D model of the durian, repeat what we did for the apple
  - not as smooth as an apple
  - the 3D model becomes very complicated
- Wait, what is a model again?

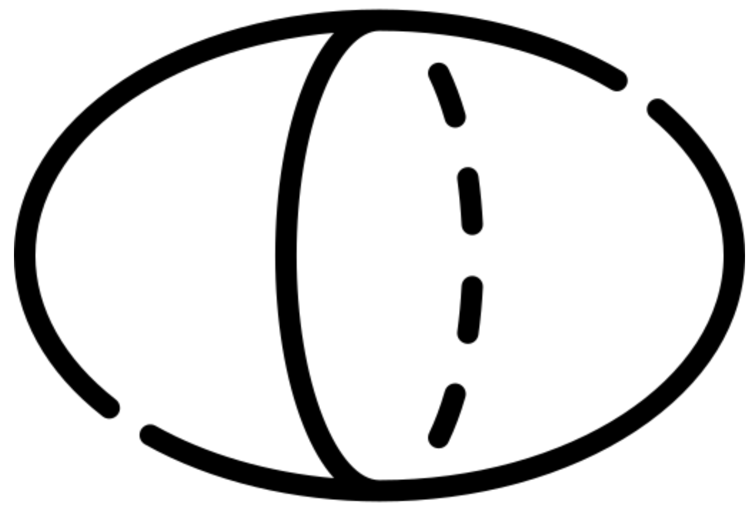


Sphere and Ellipsoid Models

Sphere: one parameter (radius)

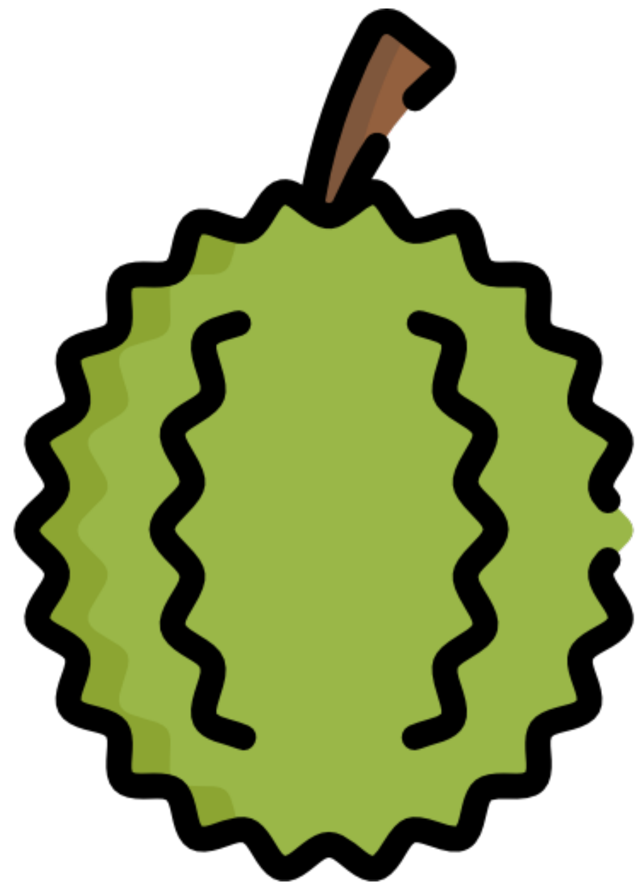


Ellipsoid: three parameters



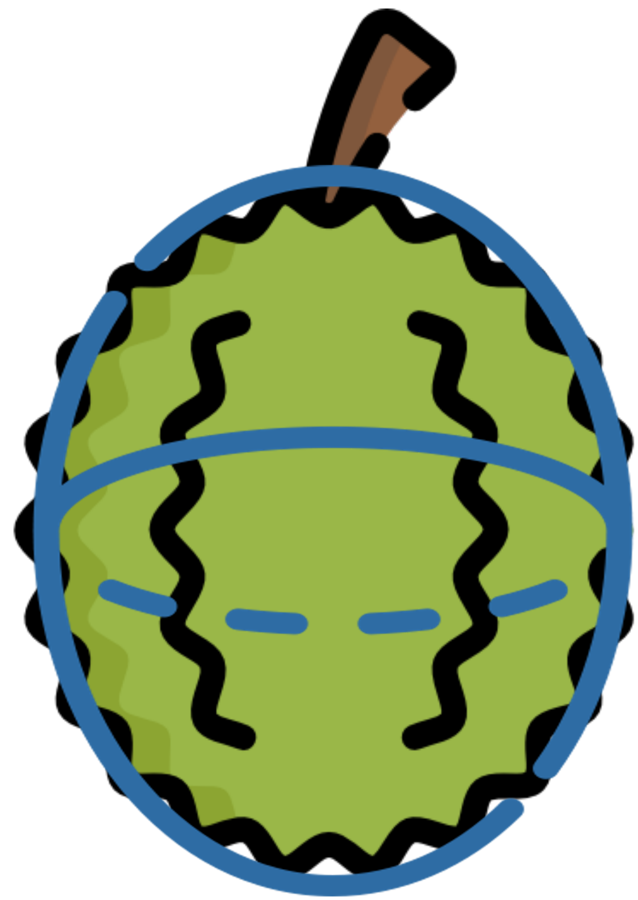
Sphere icons created by Nadiinko - Flaticon
Ellipsoid icons created by Freepik - Flaticon

An ellipsoid model is easier to deal with than a durian…

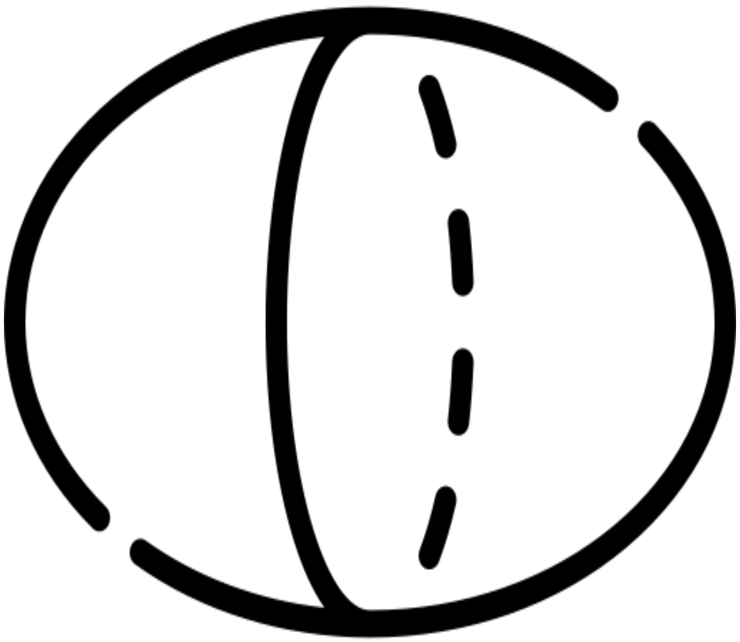


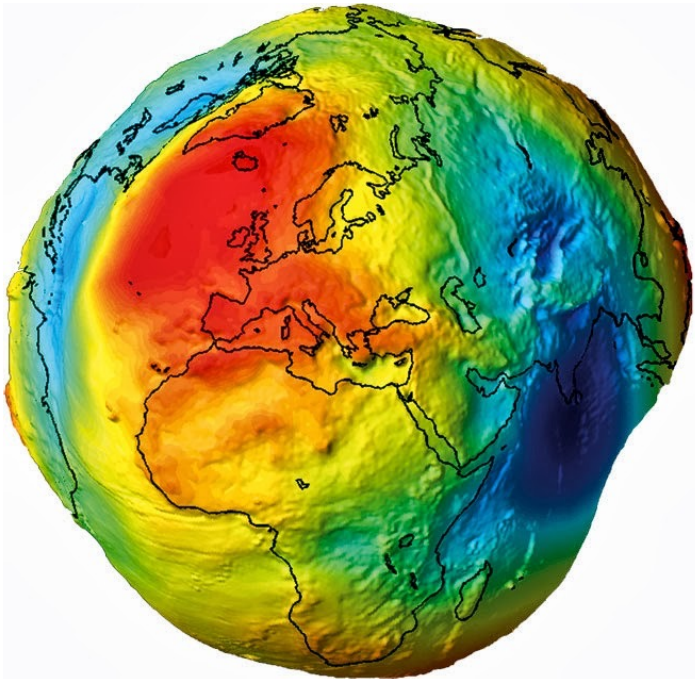
An ellipsoid model is easier to deal with than a durian…*

- Let’s find the “best” sphere to approximate our durian
- Use the sphere to talk about locations on the durian



We can do the same for the Earth



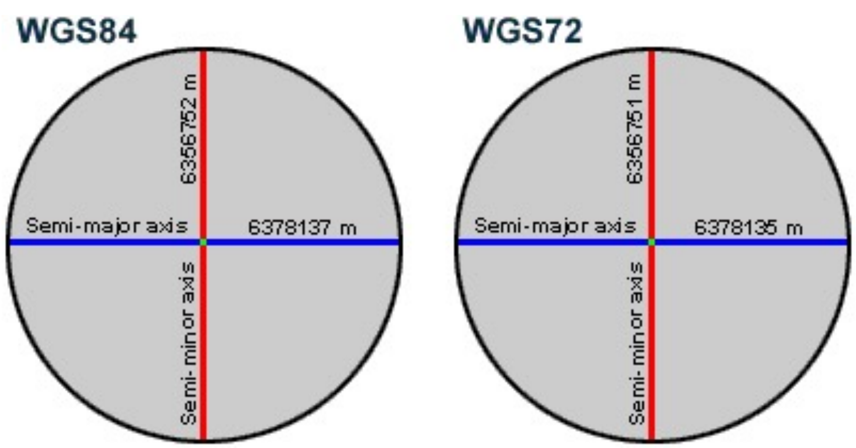


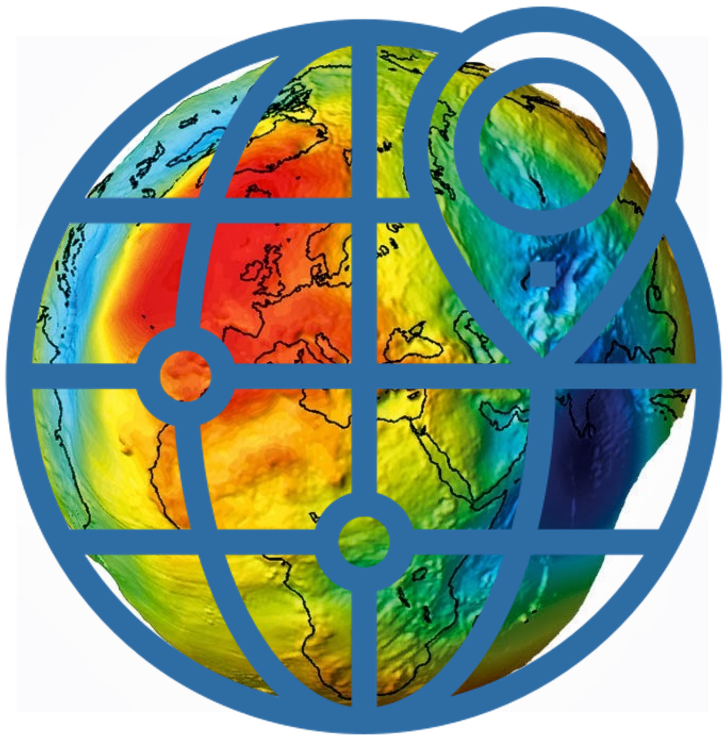
Geoid: a math figure of the Earth

https://en.wikipedia.org/wiki/Earth_Gravitational_Model

We can do the same for modeling Earth

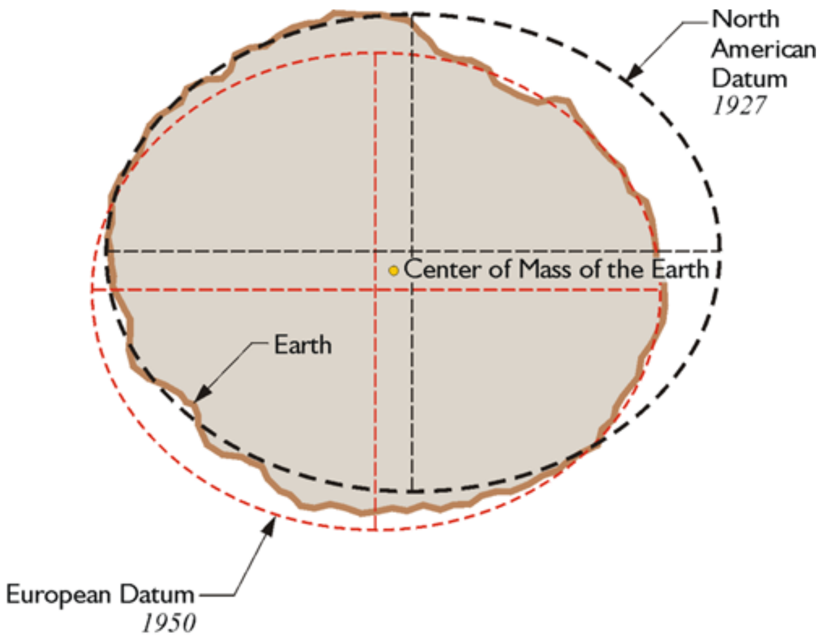
- Problem 1:
  - How to decide the best ellipsoid?
    - i.e., how to select the parameters for the ellipsoid

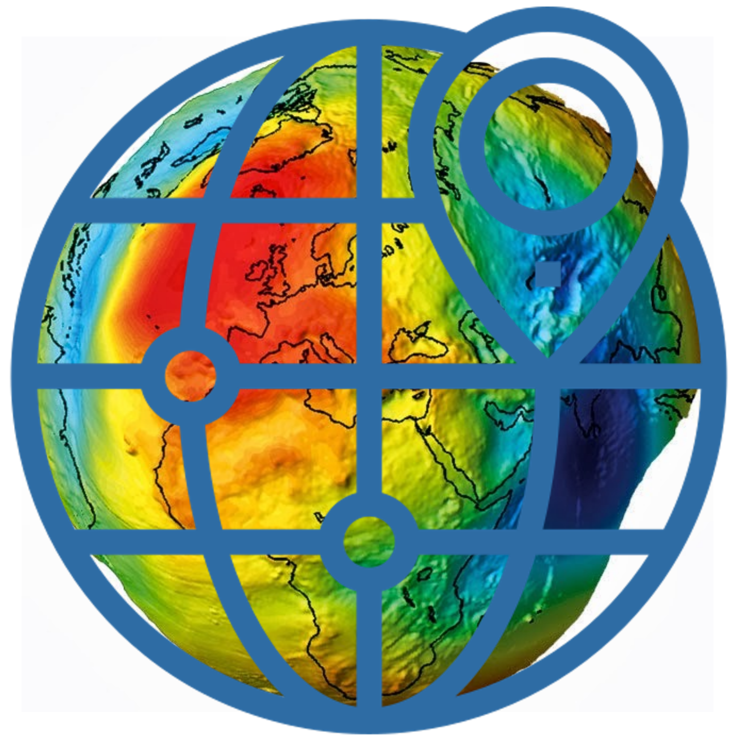




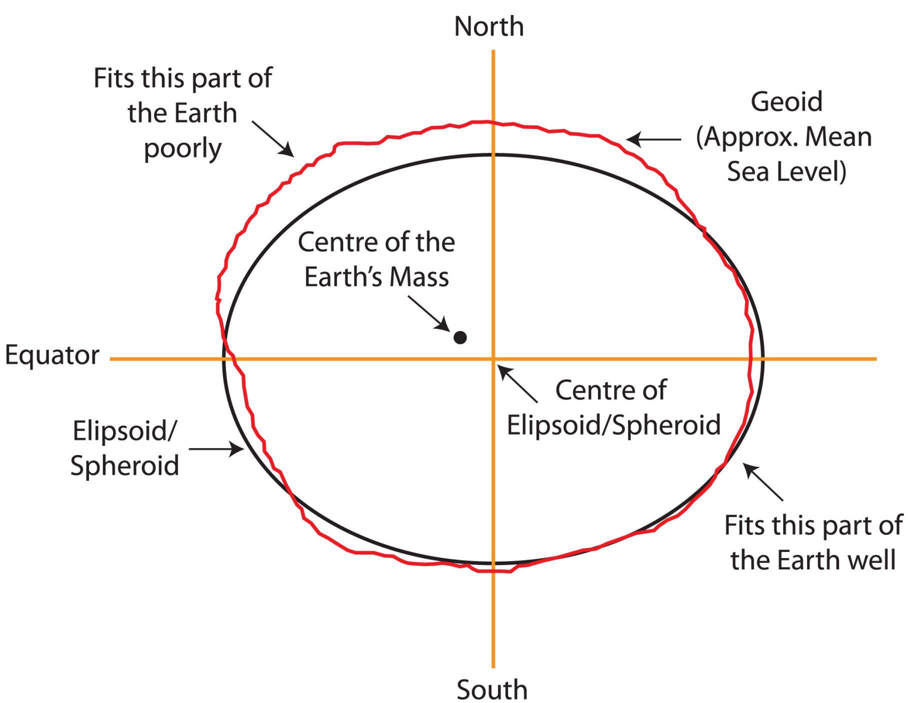
We can do the same for modeling Earth

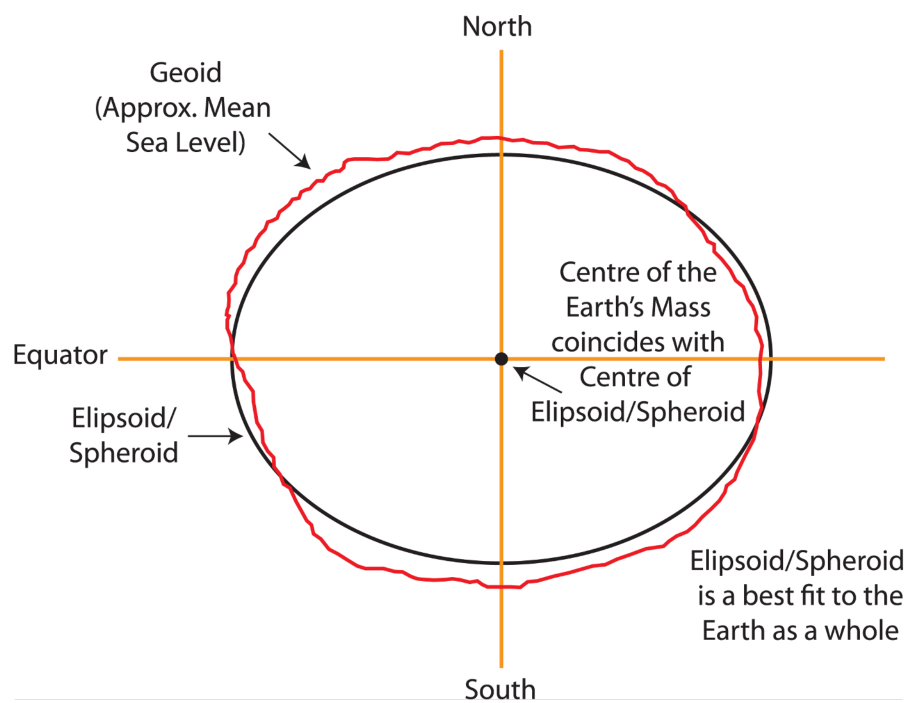
- Problem 2:
  - How to align the ellipsoid to Earth?





Datum

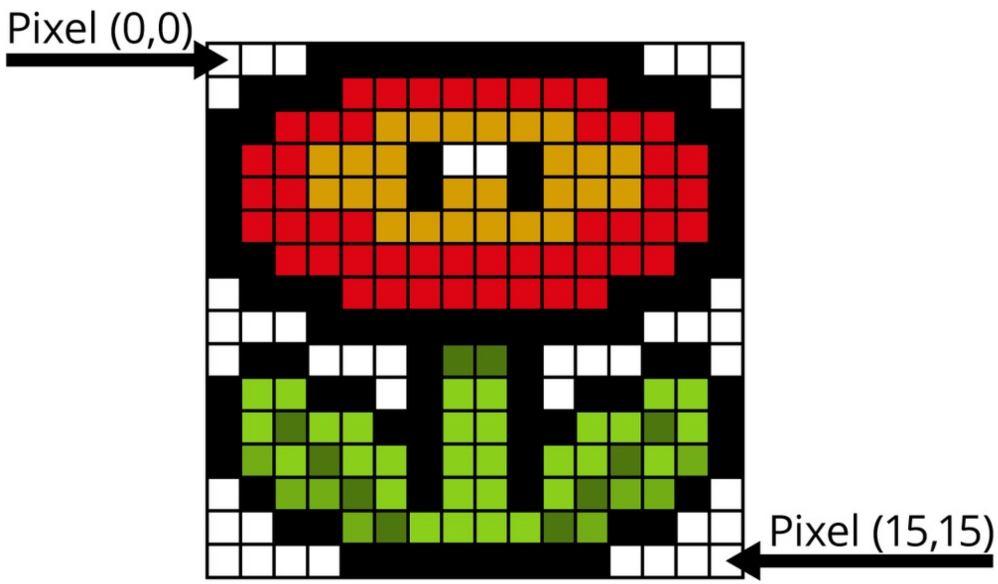




https://www.icsm.gov.au/education/fundamentals-mapping/datums

Spatial Reference Systems

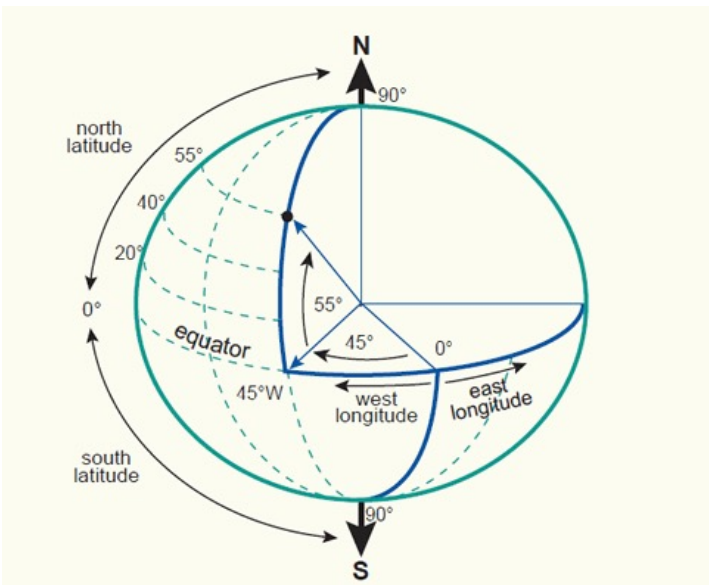
- A standardized method for assigning codes to locations so that locations can be found using the codes alone.
- In a geospatial coordinate system, the x-direction value is the easting, and the y-direction value is the northing.
- "Most" systems make both values positive.



https://www.oreilly.com/library/view/practical-computer-vision/9781449337865/ch04.html

Geographic Coordinates

- Geographic coordinates are the Earth's latitude and longitude system
  - ranging from 90 degrees south to 90 degrees north in latitude and 180 degrees west to 180 degrees east in longitude



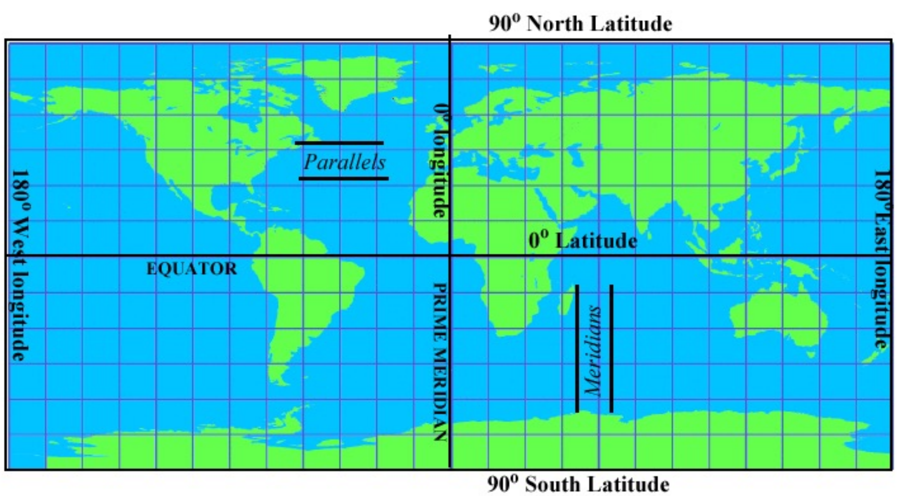
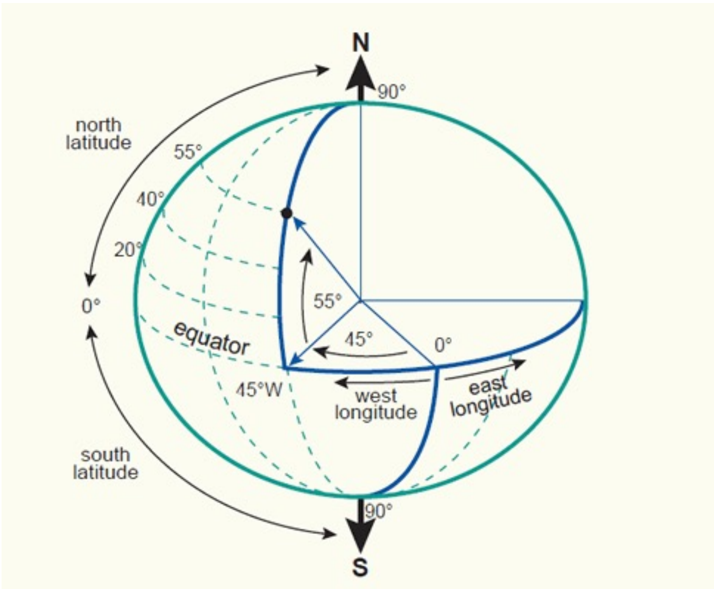


Figure 2.6 Geographic coordinates. The familiar latitude and longitude system, simply converting the angles at the earth's center to coordinates, gives the basic equirectangular projection. The map is twice as wide as high (360° east-west, 180° north-south).

Geographic Coordinates

Wait, how do we go from 3D ellipsoid to 2D maps?



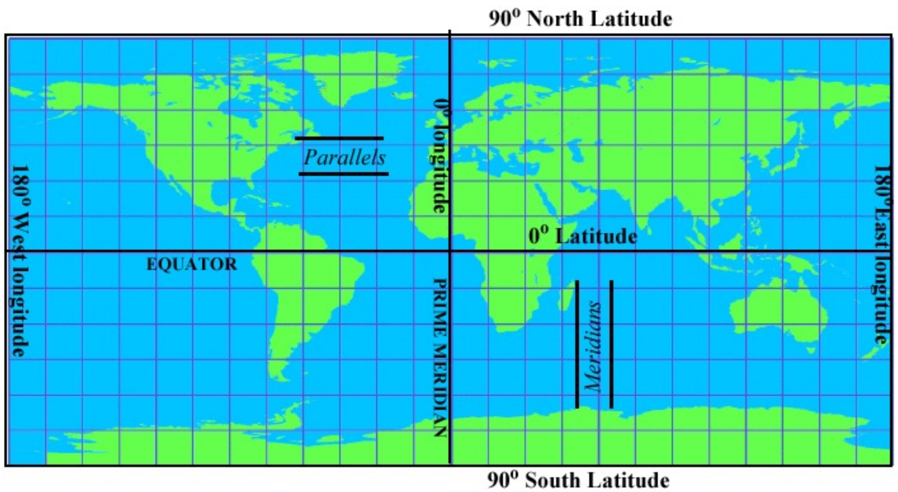
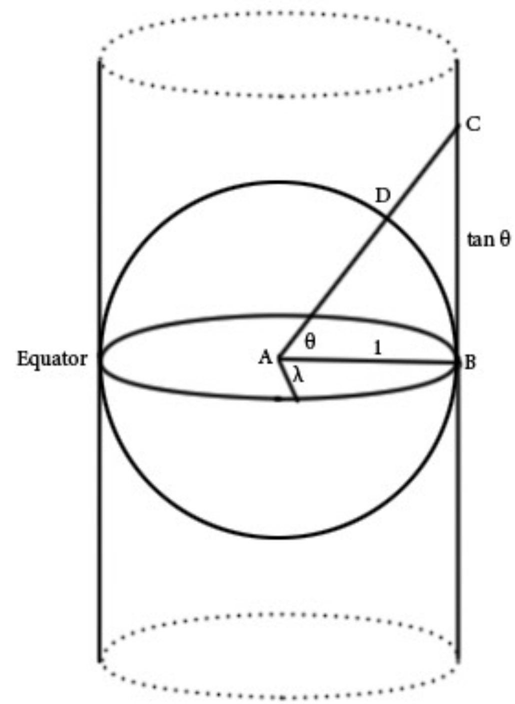


Figure 2.6 Geographic coordinates. The familiar latitude and longitude system, simply converting the angles at the earth's center to coordinates, gives the basic equirectangular projection. The map is twice as wide as high (360° east-west, 180° north-south).

Map Projections

- A mathematical transformation of the spherical or ellipsoidal Earth onto a flat map
- A projection that preserves the shape of features across the map is called conformal.
- A projection that preserves the area of a feature across the map is called equal area or equivalent.
- **No flat map can be both equivalent and conformal. Most fall between the two as compromises.**



Universal Transverse Mercator (UTM)

- Project a small piece of the 3D Earth to a 2D map
- Small grids help compute distances and directions easily

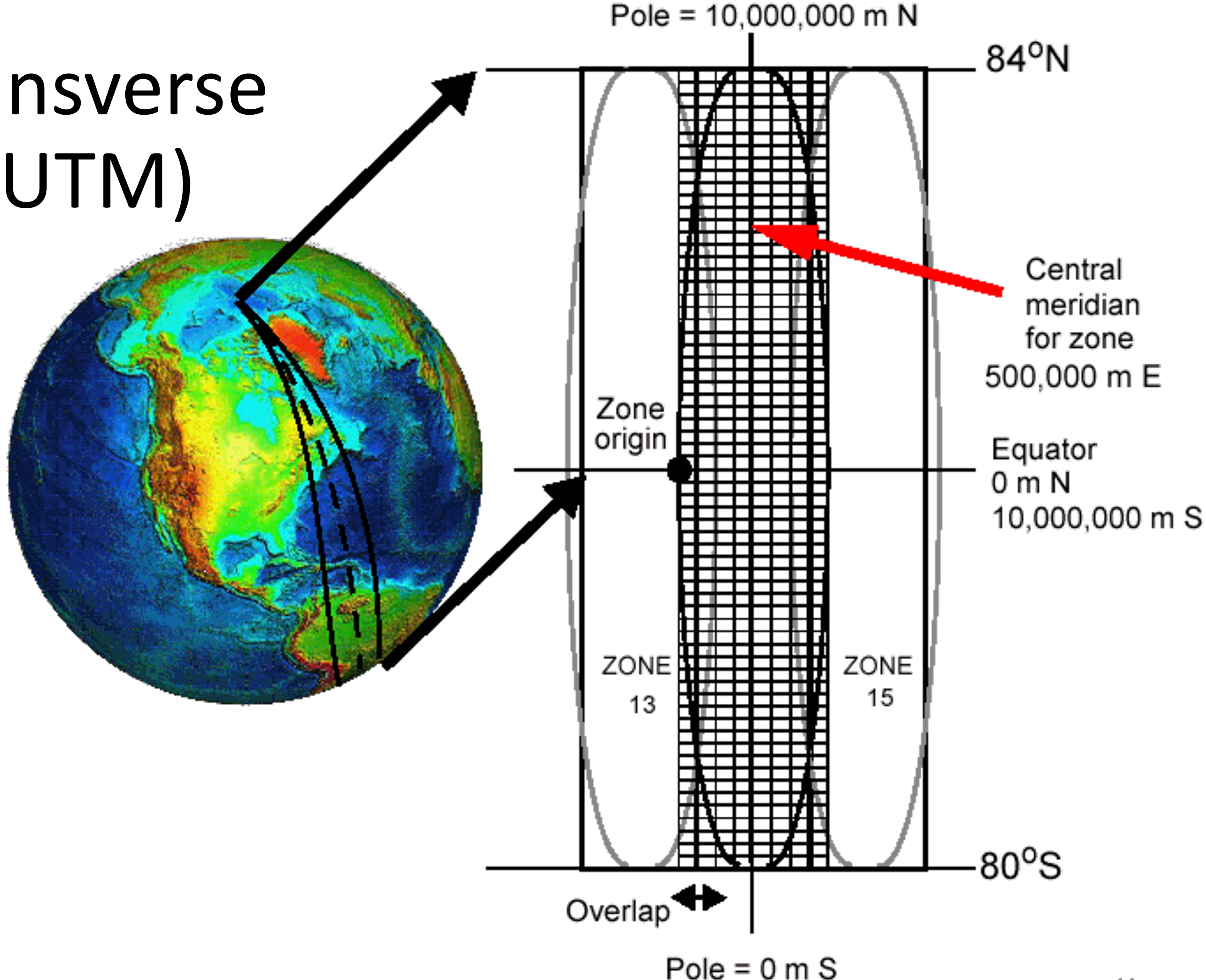


Figure 2.14 The universal transverse Mercator coordinate system.

Spatial Reference System Identifier (SRID)

Home | Upload Your Own | List user-contributed references | List all references

Previous: EPSG:2269: NAD83 / Oregon North (ft) | Next: EPSG:2271: NAD83 / Pennsylvani

Input Coordinates: -12
4857353.371193, 58

EPSG:2270

NAD83 / Oregon South (ft) (Google it)

- **WGS84 Bounds**: -124.6000, 42.0000, -116.8800, 44.5600
- **Projected Bounds**: 3807130.6123, 142752.9914, 5905041.0689, 1080736.3326
- **Scope**: Large and medium scale topographic mapping and engineering survey.
- **Last Revised**: Nov. 6, 2001
- **Area**: USA - Oregon - SPCS - S

- Well Known Text as HTML
- Human-Readable OGC WKT
- Proj4
- OGC WKT
- JSON
- GML
- ESRI WKT
- .PRJ File
- USGS
- MapServer Mapfile | Python
- Mapnik XML | Python
- GeoServer
- PostGIS spatial_ref_sys INSERT statement
- Proj4js format

https://spatialreference.org/ref/

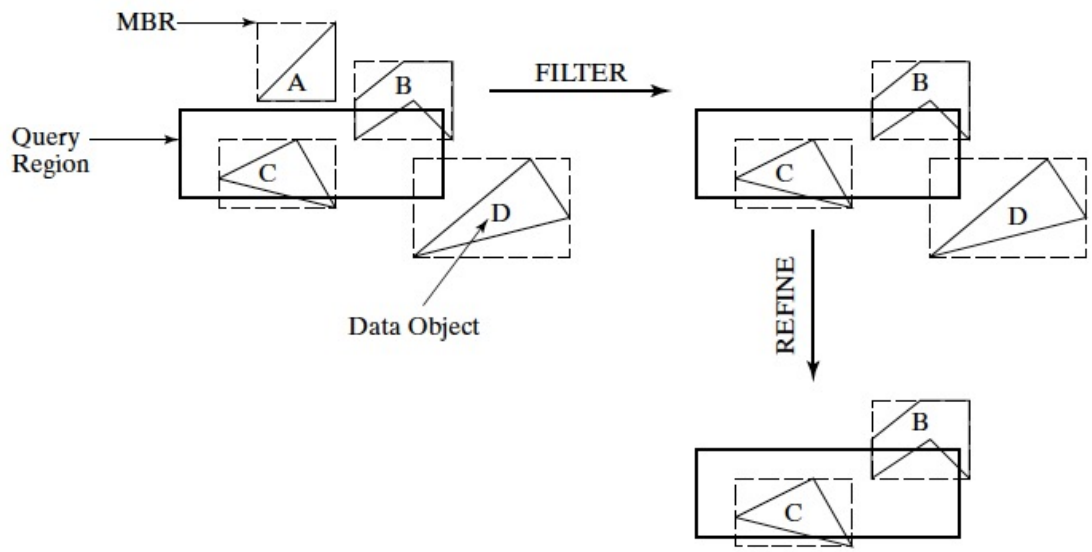
Spatial Query Language

- Spatial data types, e.g., point, lines, polygon, …
- Spatial operations, e.g., overlap, distance, nearest neighbor, …
- Callable from a query language (e.g., SQL) of underlying DBMS

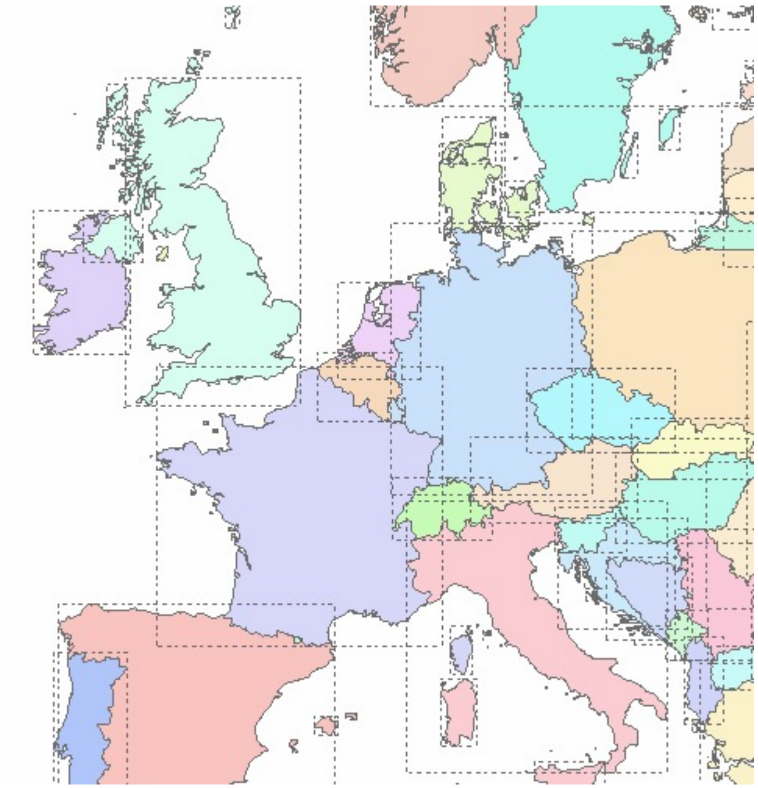
```
SELECT S.name
FROM Senator S
WHERE S.district.Area() > 300
```

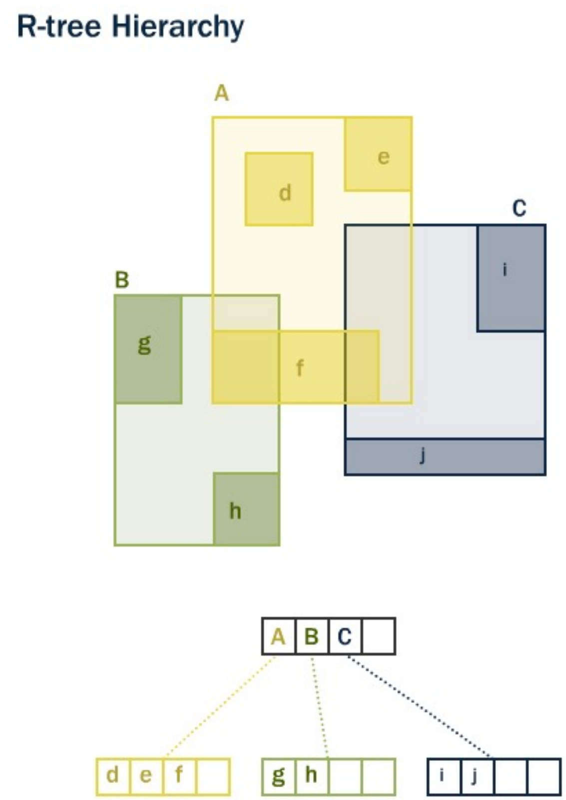
Spatial Index and Query Processing

- Efficient algorithms to answer spatial queries
- Common Strategy - filter and refine
  - Filter Step: Query Region overlaps with MBRs of B,C and D
  - Refine Step: Query Region overlaps with B and C



Spatial Index and Query Processing – R-tree





Example Queries

- Fundamental spatial algebra operations
  - Spatial selection: returning those objects satisfying a spatial predicate with the query object
    - "All cities in Bavaria"

SELECT sname FROM cities c WHERE c.center inside Bavaria.area

- "All rivers intersecting a query window"

SELECT * FROM rivers r WHERE r.route intersects Window

- "All big cities no more than 100 Kms from Hagen"

SELECT cname FROM cities c WHERE dist(c.center, Hagen.center) < 100 and c.pop > 500k

(conjunction with other predicates and query optimization)

Example Queries…*

- *Spatial join: A join which compares any* two joined objects based on a predicate on their spatial attribute values
  - “For each river pass through Bavaria, find all cities within less than 50 Km”

SELECT

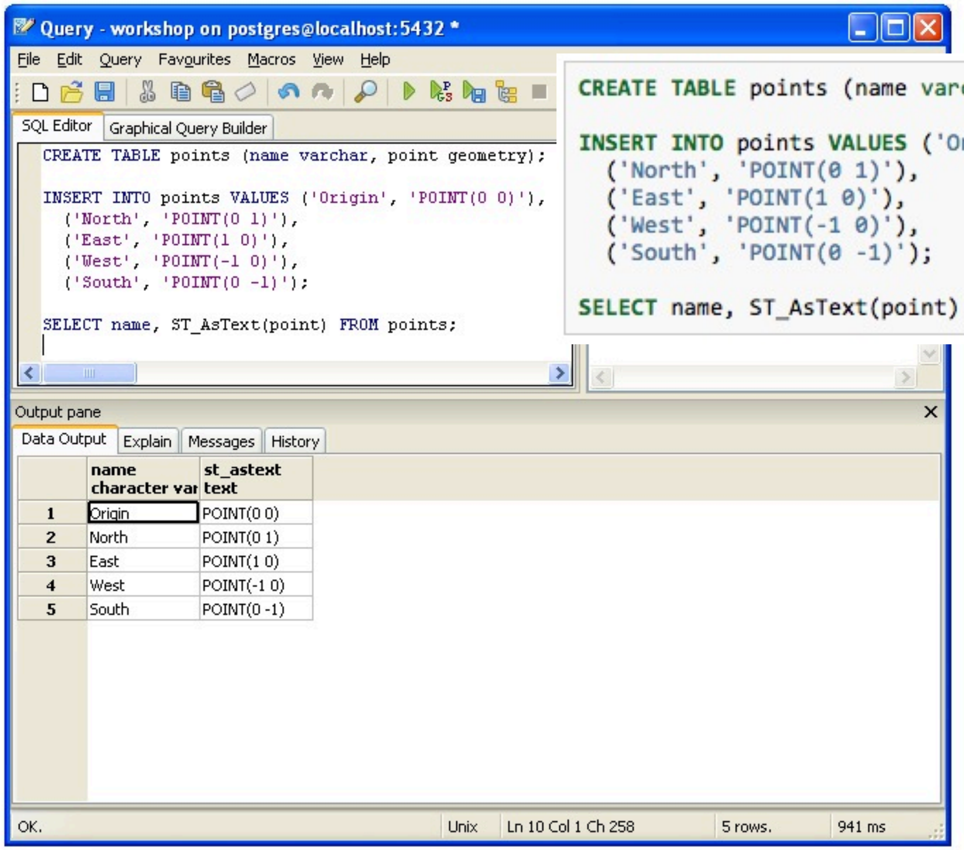
r.rname, c.cname, length(intersection(r.route, c.area))

FROM rivers r, cities c

WHERE r.route intersects Bavaria.area and

dist(r.route,c.area) < 50 Km

SFSQL Example



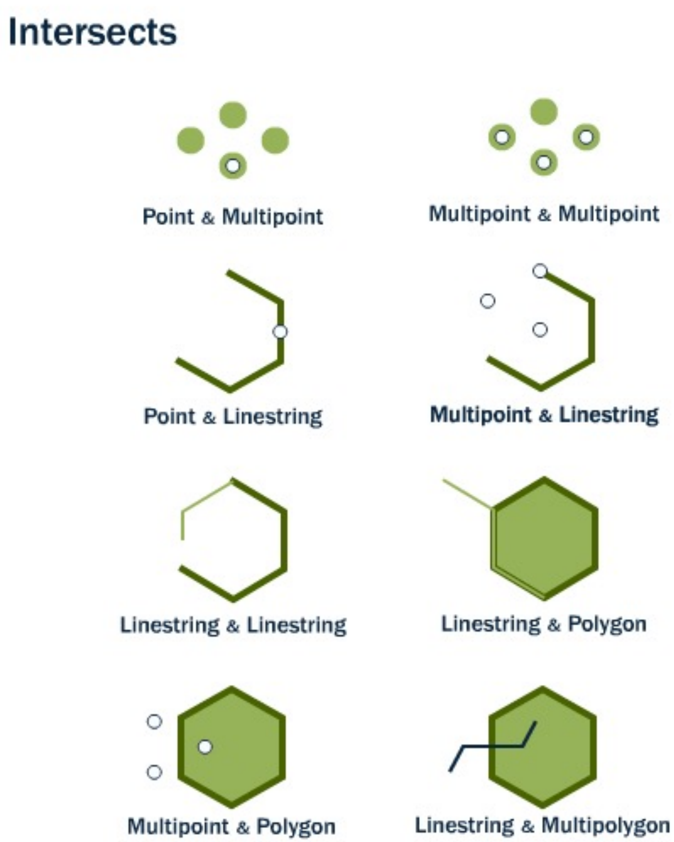

```sql
CREATE TABLE points (name varchar, point geometry);

INSERT INTO points VALUES ('Origin', 'POINT(0 0)'),
  ('North', 'POINT(0 1)'),
  ('East', 'POINT(1 0)'),
  ('West', 'POINT(-1 0)'),
  ('South', 'POINT(0 -1)');

SELECT name, ST_AsText(point) FROM points;
```

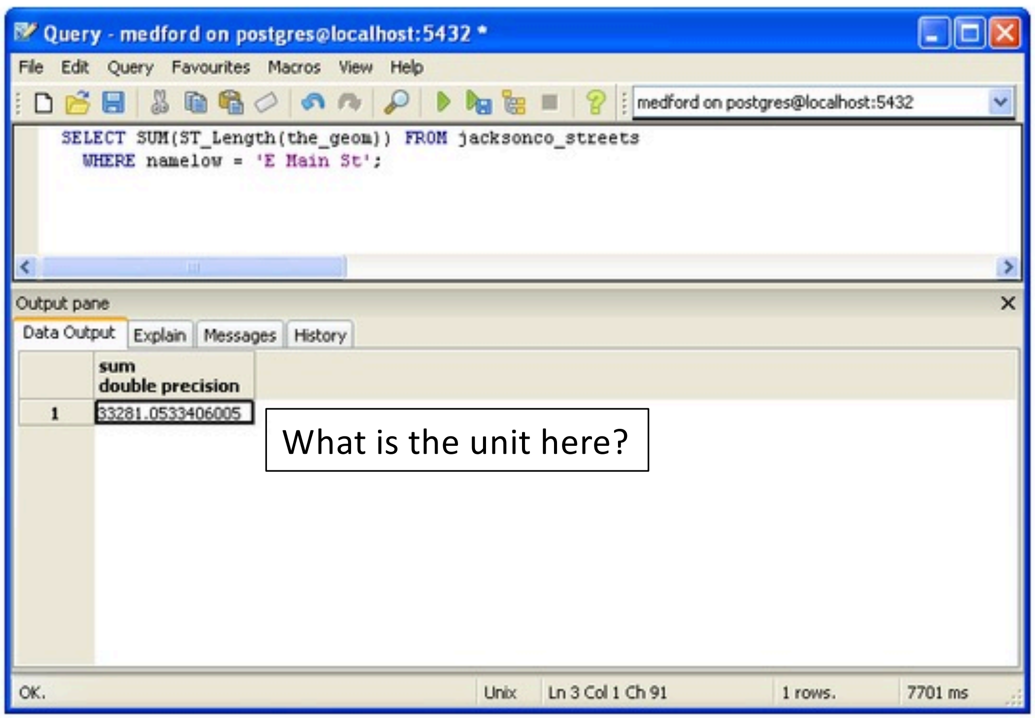
ST_Intersects

ST_Intersects(geometry A, geometry B) returns t (TRUE) if the two shapes have any space in common, i.e., if their boundaries or interiors intersect.



Query with SRID in Mind

```
SELECT SUM(ST_Length(the_geom))
  FROM jacksonco_streets
  WHERE namelow = 'E Main St';
```

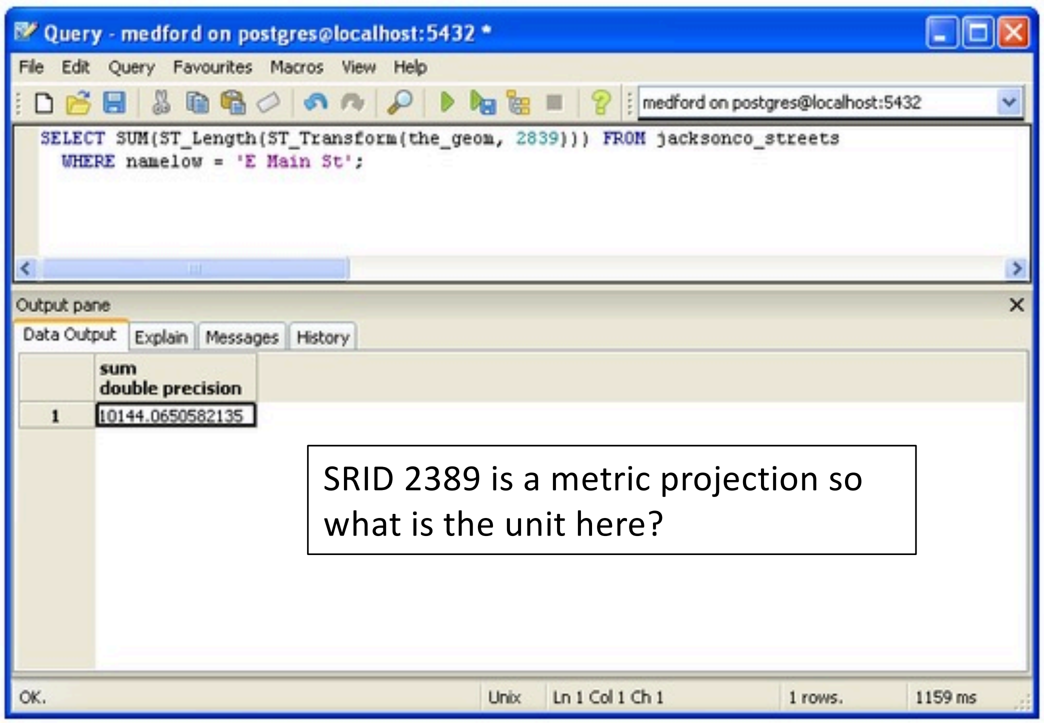


What is the unit here?

The table "jacksonco_streets" has its SRID as 2270 (EPSG:**2270**: NAD83 / Oregon South (ft))

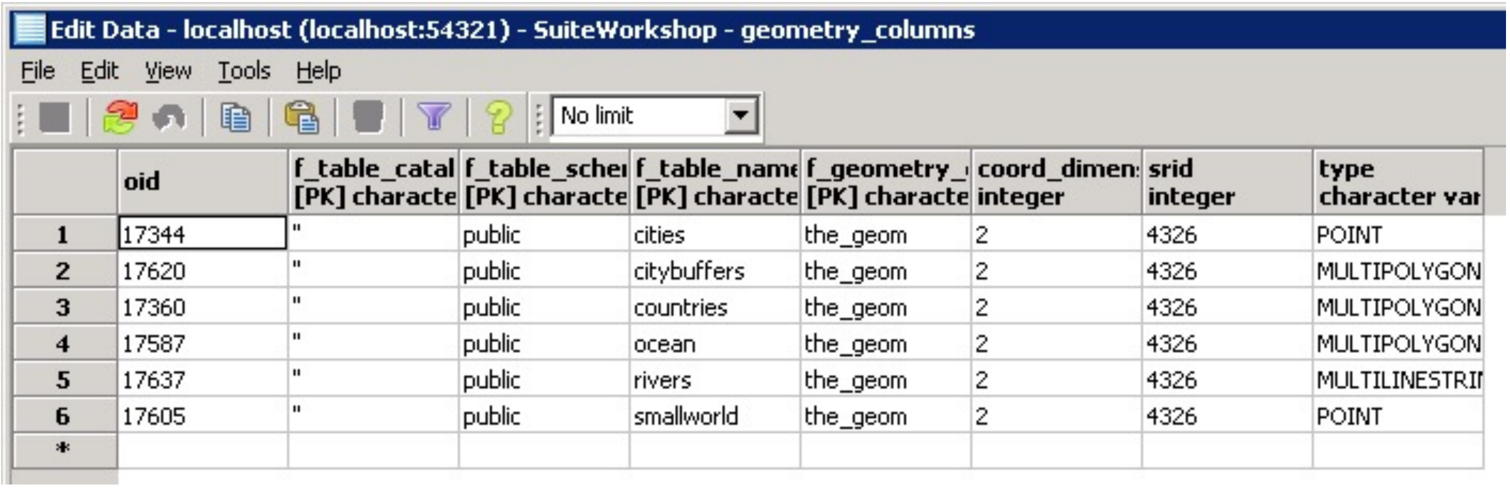
Query with SRID in Mind*

```sql
SELECT SUM(ST_Length(ST_Transform(the_geom, 2839)))
  FROM jacksonco_streets
  WHERE namelow = 'E Main St';
```



SRID 2389 is a metric projection so what is the unit here?

Geometry Columns



Edit Data - localhost (localhost:54321) - SuiteWorkshop - geometry_columns

File Edit View Tools Help

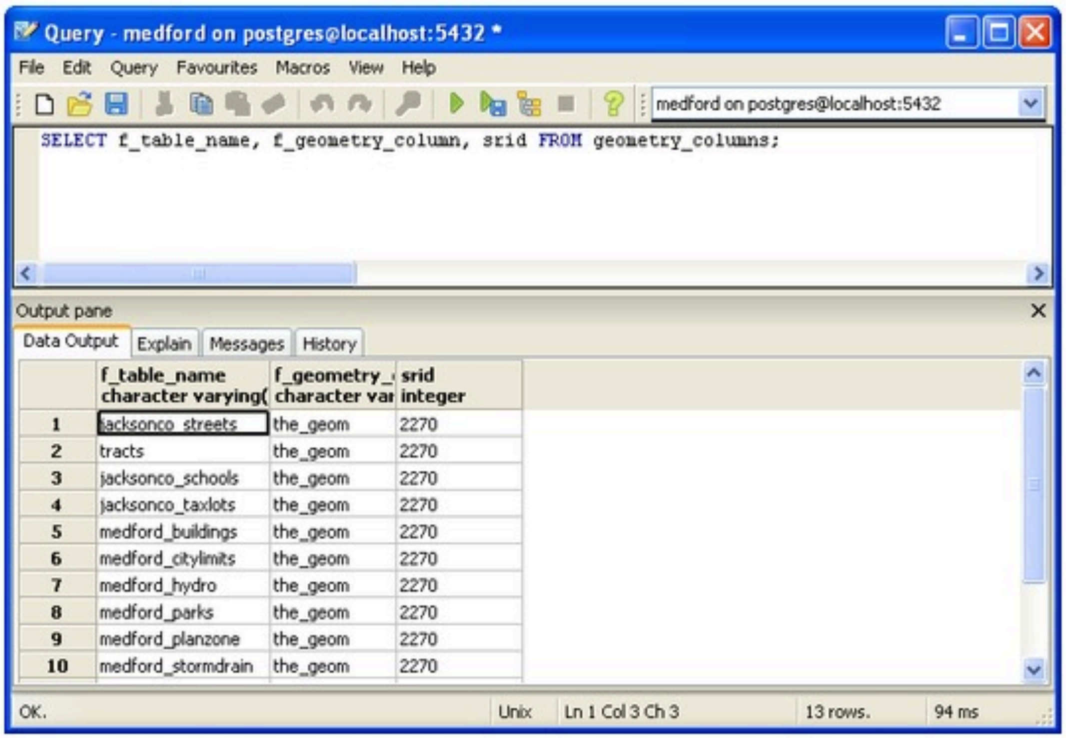
No limit

	oid	f_table_catal [PK] characte	f_table_schei [PK] characte	f_table_name [PK] characte	f_geometry_ [PK] characte	coord_dimen: integer	srid integer	type character var
1	17344	''	public	cities	the_geom	2	4326	POINT
2	17620	''	public	citybuffers	the_geom	2	4326	MULTIPOLYGON
3	17360	''	public	countries	the_geom	2	4326	MULTIPOLYGON
4	17587	''	public	ocean	the_geom	2	4326	MULTIPOLYGON
5	17637	''	public	rivers	the_geom	2	4326	MULTILINESTRI
6	17605	''	public	smallworld	the_geom	2	4326	POINT
*								

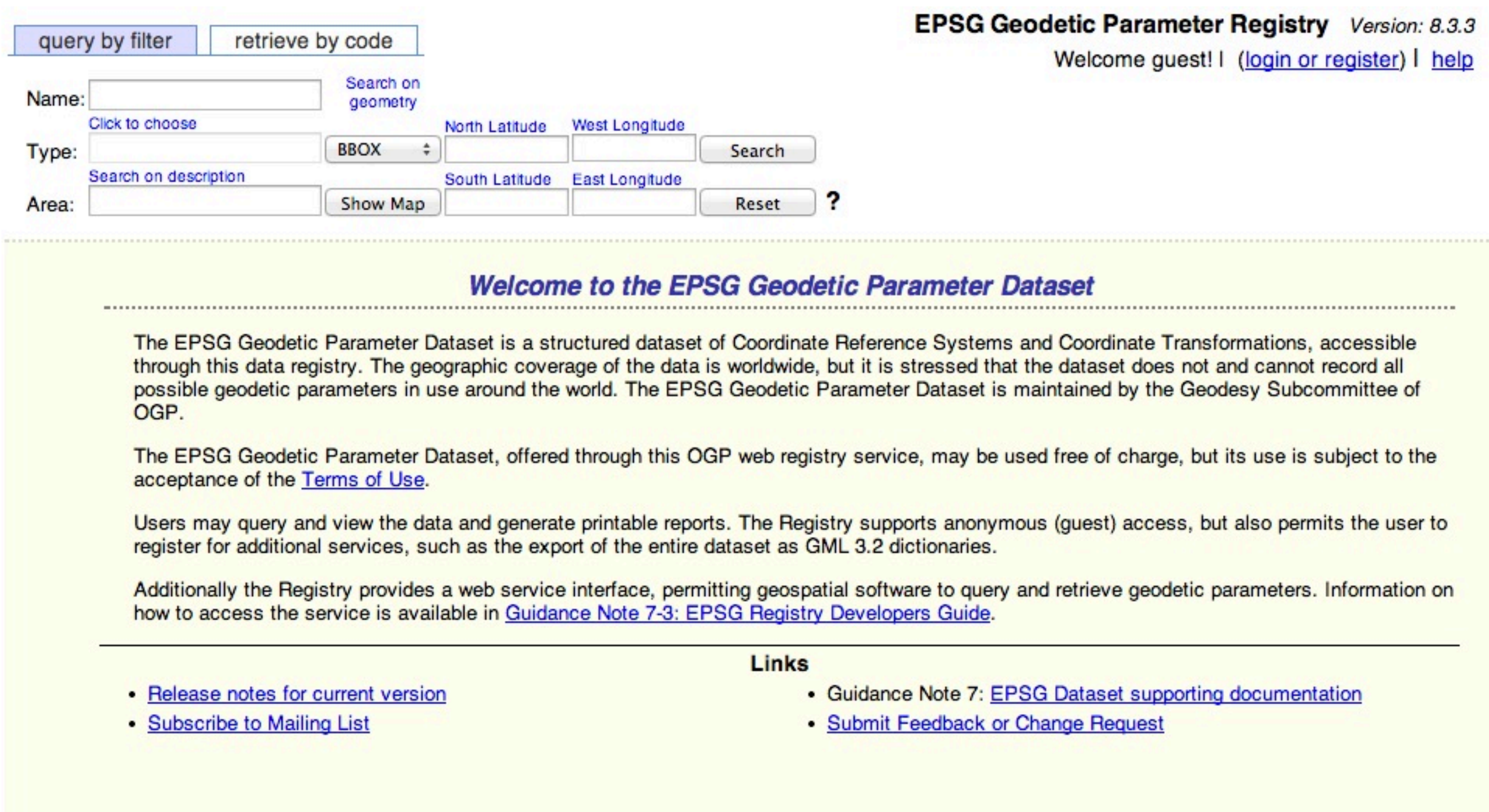
The geometry_columns view defines the dimension, geometry, and spatial reference system for each spatial table that contains a geometry type

Geometry Columns*

```sql
SELECT f_table_name, f_geometry_column, srid
  FROM geometry_columns;
```



EPSG Registry



query by filter | retrieve by code

EPSG Geodetic Parameter Registry *Version: 8.3.3*

Welcome guest! | (login or register) | help

Name: | Search on geometry

Click to choose | North Latitude | West Longitude

Type: | BBOX | Search

Search on description | South Latitude | East Longitude

Area: | Show Map | Reset ?

Welcome to the EPSG Geodetic Parameter Dataset

The EPSG Geodetic Parameter Dataset is a structured dataset of Coordinate Reference Systems and Coordinate Transformations, accessible through this data registry. The geographic coverage of the data is worldwide, but it is stressed that the dataset does not and cannot record all possible geodetic parameters in use around the world. The EPSG Geodetic Parameter Dataset is maintained by the Geodesy Subcommittee of OGP.

The EPSG Geodetic Parameter Dataset, offered through this OGP web registry service, may be used free of charge, but its use is subject to the acceptance of the Terms of Use.

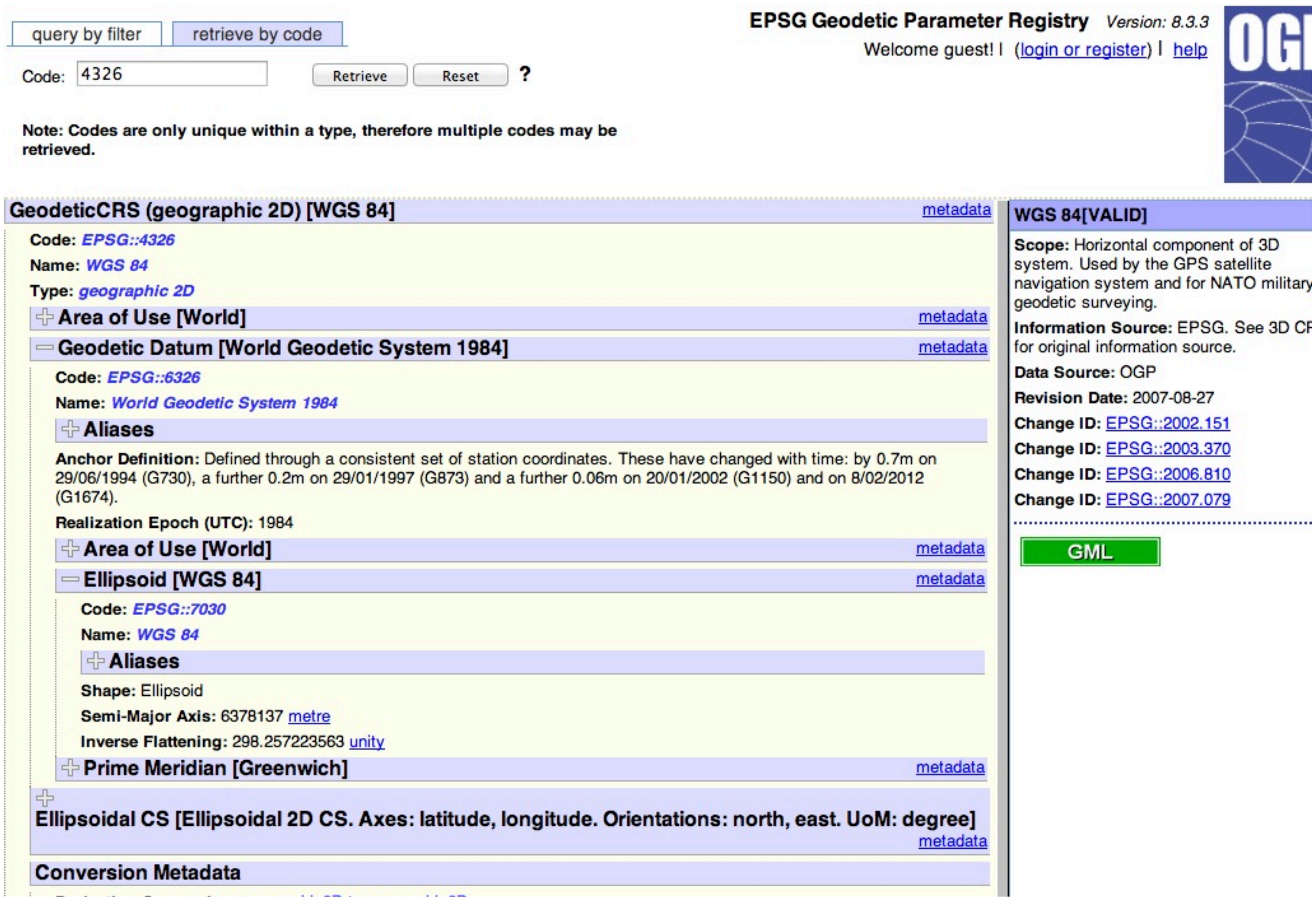
Users may query and view the data and generate printable reports. The Registry supports anonymous (guest) access, but also permits the user to register for additional services, such as the export of the entire dataset as GML 3.2 dictionaries.

Additionally the Registry provides a web service interface, permitting geospatial software to query and retrieve geodetic parameters. Information on how to access the service is available in Guidance Note 7-3: EPSG Registry Developers Guide.

Links

- Release notes for current version
- Subscribe to Mailing List
- Guidance Note 7: EPSG Dataset supporting documentation
- Submit Feedback or Change Request

EPSG Registry*



Coordinate System Support in Spatial Databases: spatial_ref_sys

	srid [PK] integer	auth_name character v	auth_srid integer	srtext character varying(2048)	proj4text character varying(2048)
1	2000	EPSG	2000	PROJCS["Anguilla 1957 / British West Indie	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.99950000
2	2001	EPSG	2001	PROJCS["Antigua 1943 / British West Indies	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.99950000
3	2002	EPSG	2002	PROJCS["Dominica 1945 / British West Indie	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.99950000
4	2003	EPSG	2003	PROJCS["Grenada 1953 / British West Indies	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.99950000
5	2004	EPSG	2004	PROJCS["Montserrat 1958 / British West Ind	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.99950000
6	2005	EPSG	2005	PROJCS["St. Kitts 1955 / British West Indi	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.99950000
7	2006	EPSG	2006	PROJCS["St. Lucia 1955 / British West Indi	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.99950000
8	2007	EPSG	2007	PROJCS["St. Vincent 45 / British West Indi	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.99950000
9	2008	EPSG	2008	PROJCS["NAD27(CGQ77) / SCoPQ zone 2",GEOGC	+proj=tmerc +lat_0=0 +lon_0=-55.5 +k=0.9999
10	2009	EPSG	2009	PROJCS["NAD27(CGQ77) / SCoPQ zone 3",GEOGC	+proj=tmerc +lat_0=0 +lon_0=-58.5 +k=0.9999
11	2010	EPSG	2010	PROJCS["NAD27(CGQ77) / SCoPQ zone 4",GEOGC	+proj=tmerc +lat_0=0 +lon_0=-61.5 +k=0.9999
12	2011	EPSG	2011	PROJCS["NAD27(CGQ77) / SCoPQ zone 5",GEOGC	+proj=tmerc +lat_0=0 +lon_0=-64.5 +k=0.9999
13	2012	EPSG	2012	PROJCS["NAD27(CGQ77) / SCoPQ zone 6",GEOGC	+proj=tmerc +lat_0=0 +lon_0=-67.5 +k=0.9999
14	2013	EPSG	2013	PROJCS["NAD27(CGQ77) / SCoPQ zone 7",GEOGC	+proj=tmerc +lat_0=0 +lon_0=-70.5 +k=0.9999
15	2014	EPSG	2014	PROJCS["NAD27(CGQ77) / SCoPQ zone 8",GEOGC	+proj=tmerc +lat_0=0 +lon_0=-73.5 +k=0.9999
16	2015	EPSG	2015	PROJCS["NAD27(CGQ77) / SCoPQ zone 9",GEOGC	+proj=tmerc +lat_0=0 +lon_0=-76.5 +k=0.9999
17	2016	EPSG	2016	PROJCS["NAD27(CGQ77) / SCoPQ zone 10",GEOG	+proj=tmerc +lat_0=0 +lon_0=-79.5 +k=0.9999
18	2017	EPSG	2017	PROJCS["NAD27(76) / MTM zone 8",GEOGCS["NA	+proj=tmerc +lat_0=0 +lon_0=-73.5 +k=0.9999

Every Geometry Column is associated with a Spatial Reference System

Geography VS Geometry

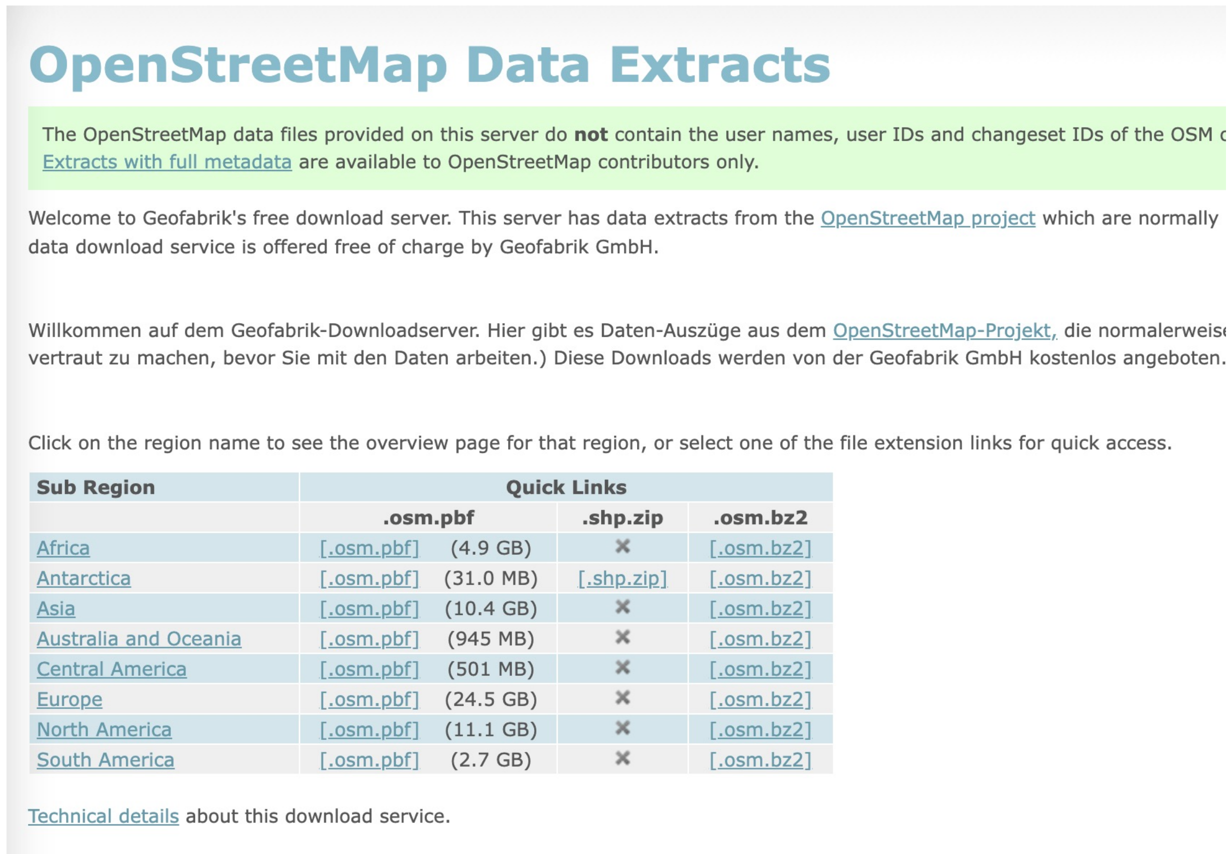
- The Geometry column type can hold geometric data of any type and in any (or no) projection and CRS.
  - not optimized for dealing with geodetic measurements (distances on the sphere)
- The Geography type, (while able to handle geodetic measurements), are much more limited
  - there are fewer compatible functions when compared to Geometry

Summary

- Effective spatial data management strategies can enable many spatial AI tasks
- SDBMS is a software module
  - works with an underlying DBMS
  - provides spatial ADTs callable from a query language
  - provides methods for efficient processing of spatial queries (e.g., using index)
- Components of SDBMS include
  - spatial data models, spatial data types, and operators
  - information about common spatial reference systems
  - spatial query language, processing, and optimization
- Always handle spatial data with reference systems in mind!

Spatial Data Can be Huge

- How to deal with large spatial data?
    - Next time - MapReduce



OpenStreetMap Data Extracts

The OpenStreetMap data files provided on this server do **not** contain the user names, user IDs and changeset IDs of the OSM c
Extracts with full metadata are available to OpenStreetMap contributors only.

Welcome to Geofabrik's free download server. This server has data extracts from the OpenStreetMap project which are normally
data download service is offered free of charge by Geofabrik GmbH.

Willkommen auf dem Geofabrik-Downloadserver. Hier gibt es Daten-Auszüge aus dem OpenStreetMap-Projekt, die normalerweis
vertraut zu machen, bevor Sie mit den Daten arbeiten.) Diese Downloads werden von der Geofabrik GmbH kostenlos angeboten.

Click on the region name to see the overview page for that region, or select one of the file extension links for quick access.

Sub Region	Quick Links			
	.osm.pbf		.shp.zip	.osm.bz2
Africa	[.osm.pbf]	(4.9 GB)	✖	[.osm.bz2]
Antarctica	[.osm.pbf]	(31.0 MB)	[.shp.zip]	[.osm.bz2]
Asia	[.osm.pbf]	(10.4 GB)	✖	[.osm.bz2]
Australia and Oceania	[.osm.pbf]	(945 MB)	✖	[.osm.bz2]
Central America	[.osm.pbf]	(501 MB)	✖	[.osm.bz2]
Europe	[.osm.pbf]	(24.5 GB)	✖	[.osm.bz2]
North America	[.osm.pbf]	(11.1 GB)	✖	[.osm.bz2]
South America	[.osm.pbf]	(2.7 GB)	✖	[.osm.bz2]

Technical details about this download service.

Useful Resources

- PostGIS Workshop: https://postgis.net/workshops/postgis-intro/
- QGIS Tutorial: https://www.qgistutorials.com/en/

Acknowledgements

- Gil, Yolanda (Ed.) Introduction to Computational Thinking and Data Science. Available from http://www.datascience4all.org
- 'image: Flaticon.com'. These slides have been designed using resources from Flaticon.com
- This presentation was adapted from the slides provided from the textbook: **Spatial Databases: A Tour. Authors: Shashi Shekhar and Sanjay Chawla. Publisher: Prentice Hall, 2003,** from the database course slides provided by **Cyrus Shahabi,** from **Hart Hartmut Guting's VLDB Journal v3, n4, October 1994,** from **Boundless OpenGeo tutorial.**
- These slides are adapted from the slides provided by Keith Clarke from his course and textbook Getting Started with Geographic Information System, Prentice Hall and from Craig Knoblock