Spatial Database Management
GEP 664 / GEP 380
Class #7: Spatial reference systems

Frank Donnelly

Dept of EEGS, Lehman College CUNY

Spring 2019

Today's Topics

Spatial Reference Systems

Spatial Reference System (SRS)

Standards for modeling and locating objects on the earth that are based on geodesy (measuring and modeling the spherical earth) and cartography (representing the earth on a 2D, flat surface). All systems have 3 ingredients, some have 4.

Spatial Reference System Identifier (SRID)

A unique value that identifies a specific SRS. Within the open source GIS world, EPSG (European Petroleum Survey Group) numbers are commonly used. ESRI uses a comparable system.

Spheroid / Ellipsoid

SRS Ingredient 1

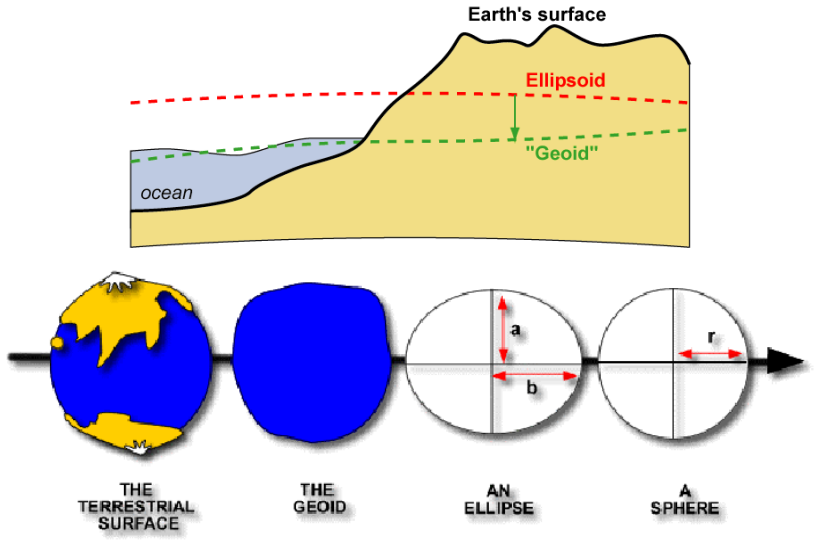


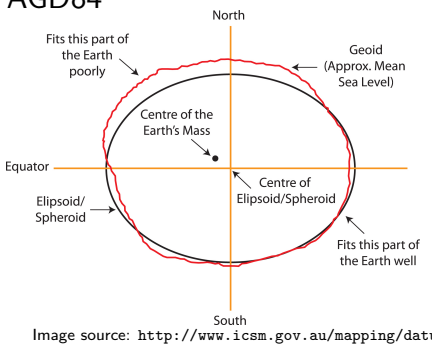
Image source: http://physics.nmsu.edu/~jni/introgeophys/05_sea_surface_and_geoid/index.html

Datums

SRS Ingredient 2

Datums are a set of instructions for attaching or anchoring ellipsoids or spheroids to the actual Earth. They can be optimized for a specific region (left) or for the earth as a whole (right). Examples below are for Australia.

AGD84



GDA94

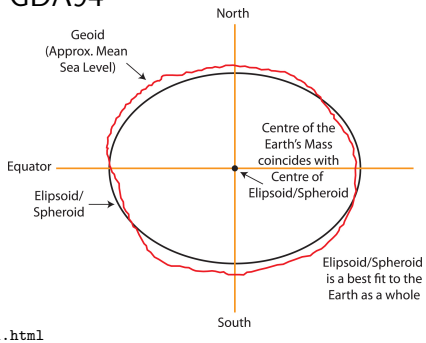


Image source: `http://www.icsm.gov.au/mapping/datums1.html`

Coordinate Reference Systems

SRS Ingredient 3

A CRS is used to identify and locate objects on the earth. The geographical coordinate system (longitude and latitude) is the most commonly used for the spherical earth and for 2D reference maps.

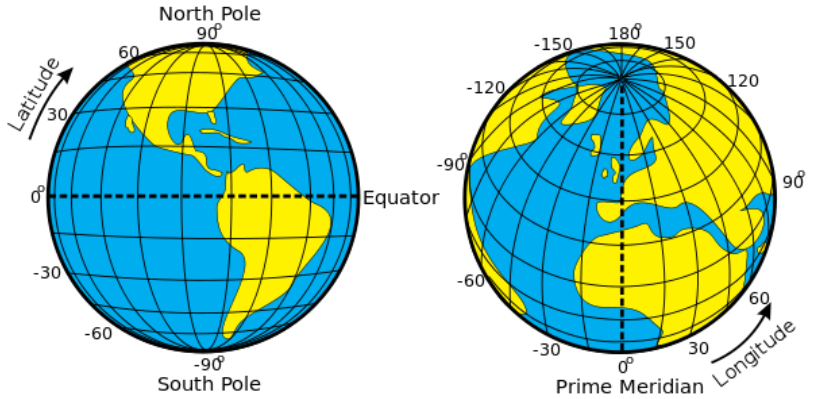


Image source: `http://www.wikiwand.com/en/Geographic_coordinate_system`

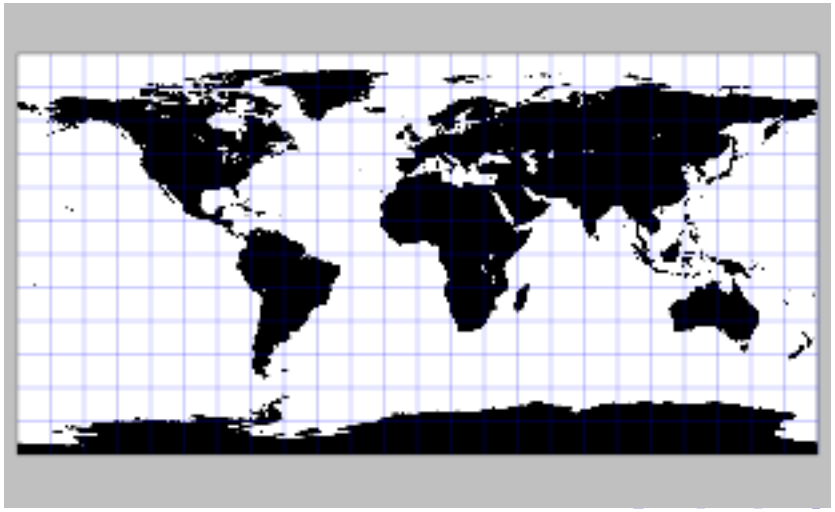
Common SRS with EPSG Codes

All of these systems use the geographic coordinate system (longitude / latitude)

Name	EPSG	Ellipsoid	Datum
WGS 84	4326	WGS 84	World Geodetic System 1984
NAD 83	4269	GRS 80	North American Datum 1983
NAD 27	4267	Clarke 1866	North American Datum 1927

Geographic Systems in 2D

With an ellipsoid, datum, and CRS you have an unprojected system used for representing the 3D earth. When displayed in 2D, all of these systems have the same appearance.



Map Projections

SRS Ingredient 4

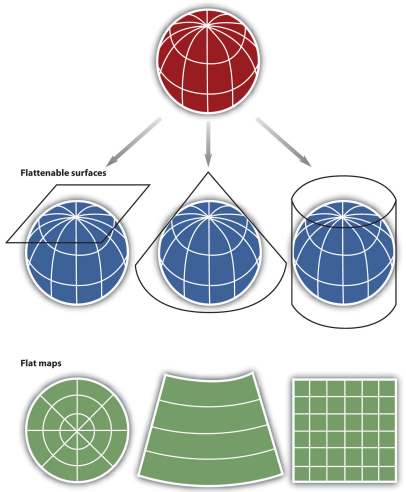
Map projections are systems used for taking the 3D earth and displaying it in 2D. Flat surface has a Cartesian (X,Y) coordinate system used for working with Euclidean geometry.

All map projections are distortions that can only preserve one property of the earth.

- Direction (azimuthal)
- Shape (conformal)
- Area (equal-area)
- Distance (equidistant)

Shape can only be preserved locally. Direction and distance can only be true between one or two points and every other point. Projections often use coordinate grids in meters or feet.

Types of Projections



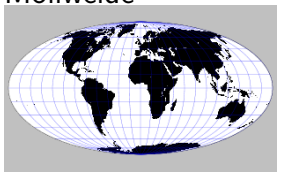
Azimuthal, Conic, and Cylindrical

Image source:
http://2012books.lardbucket.org/books/geographic-information-system-basics/s06-map-anatomy.html

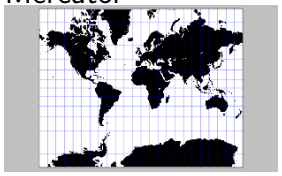
Global Systems

There are many. See:
http://www.radicalcartography.net/?projectionref

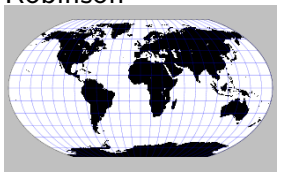
Mollweide



Mercator



Robinson



Continental Systems

Optimized for each continent. North America:

Equidistant Conic

Lambert Conformal Conic

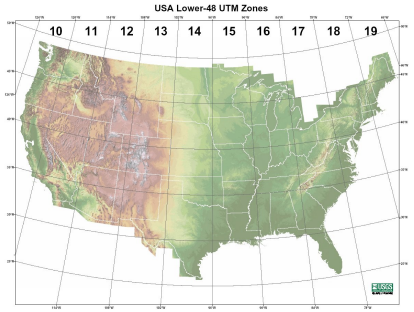
Albers Equal-Area Conic

Image source: http://www.radicalcartography.net/?projectionref

Local Systems

- Universal Transverse Mercator (UTM) divides earth into bands measuring 6 degrees north and south of equator.
- State Plane system divides US into series of zones, with larger states having multiple zones.

UTM (showing US zones)



State Plane (showing NY State)

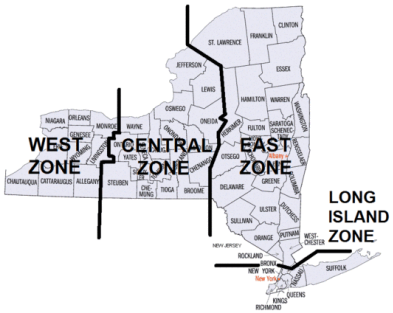


Image source: https://www.nps.gov/gis/gps/gps_info.html
Image source: http://alidade.wikispaces.com/new+york+spcs+zones

WGS 84 & NAD 83

WGS 84 is the primary system used in web mapping, freely available GIS data, and GPS data. NAD 83 is the primary system used for national datasets produced by the US and Canada.

- Good as a common interchange format
- Common in commercial web mapping
- Terrible for distance measurement
- Bad for thematic mapping and presentation

Instead of these systems, use a projected system for mapping, analysis, and local / short distance measurement.

GEOGCS and Geography type

Systems like WGS 84 and NAD 83 model the 3D earth. For measuring long distances store coordinate data in these systems in the geography data type (instead of geometry). Distance measurements in geography are geodetic and output will be converted to meters.

Today's Topics

PostGIS SRS Table

SRS definitions are stored in the spatial_ref_sys table in the public schema. Query by SRID or search the definition text with LIKE (for either GEOGCS or PROJCS).

```
SELECT *
FROM spatial_ref_sys
WHERE SRID=4269;
```

	srid integer	auth_name character varying(256)	auth_srid integer	srtext character varying(2048)	proj4text character varying(2048)
1	4269	EPSG	4269	GEOGCS["NAD83",DATUM["North_A	+proj=longlat +e

```
SELECT *
FROM spatial_ref_sys
WHERE srtext LIKE 'GEOGCS["NAD83%''
```

	srid integer	auth_name character varying(256)	auth_srid integer	srtext character varying(2048)	proj4text character varying(2048)
1	4140	EPSG	4140	GEOGCS["NAD83(CSRS98)",DATUM[	+proj=longlat +e
2	4152	EPSG	4152	GEOGCS["NAD83(HARN)",DATUM["N	+proj=longlat +e
3	4269	EPSG	4269	GEOGCS["NAD83",DATUM["North_A	+proj=longlat +e
4	4617	EPSG	4617	GEOGCS["NAD83(CSRS)",DATUM["N	+proj=longlat +e
5	4759	EPSG	4759	GEOGCS["NAD83(NSRS2007)",DATU	+proj=longlat +e

WKT - GEOGCS Example

```
GEOGCS["NAD83",
    DATUM["North_American_Datum_1983",
        SPHEROID["GRS 1980",6378137,298.257222101,
            AUTHORITY["EPSG","7019"]],
        AUTHORITY["EPSG","6269"]],
    PRIMEM["Greenwich",0,
        AUTHORITY["EPSG","8901"]],
    UNIT["degree",0.01745329251994328,
        AUTHORITY["EPSG","9122"]],
    AUTHORITY["EPSG","4269"]]
```

WKT - PROJGCS Example

```
PROJCS["NAD83 / New York Long Island (ftUS)",
    GEOGCS["NAD83",
        DATUM["North_American_Datum_1983",
            SPHEROID["GRS 1980",6378137,298.257222101,
                AUTHORITY["EPSG","7019"]],
            AUTHORITY["EPSG","6269"]],
        PRIMEM["Greenwich",0,
            AUTHORITY["EPSG","8901"]],
        UNIT["degree",0.01745329251994328,
            AUTHORITY["EPSG","9122"]],
        AUTHORITY["EPSG","4269"]],
    UNIT["US survey foot",0.3048006096012192,
        AUTHORITY["EPSG","9003"]],
    PROJECTION["Lambert_Conformal_Conic_2SP"],
    PARAMETER["standard_parallel_1",41.03333333333333],
    PARAMETER["standard_parallel_2",40.66666666666666],
    PARAMETER["latitude_of_origin",40.16666666666666],
    PARAMETER["central_meridian",-74],
    PARAMETER["false_easting",984250.0000000002],
    PARAMETER["false_northing",0],
    AUTHORITY["EPSG","2263"],
    AXIS["X",EAST],
    AXIS["Y",NORTH]]
```

Proj4

An open source library for performing conversions between SRS.

Example for a GEOGCS:

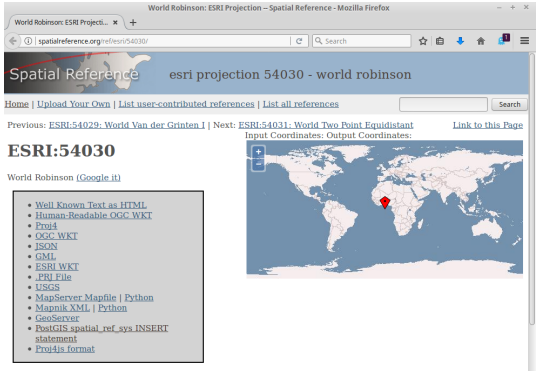
```
"+proj=longlat +ellps=GRS80
+towgs84=0,0,0,0,0,0,0 +no_defs "
```

Example for a PROJCS:

```
"+proj=lcc +lat_1=41.03333333333333
+lat_2=40.66666666666666
+lat_0=40.16666666666666
+lon_0=-74 +x_0=300000.0000000001 +y_0=0
+ellps=GRS80 +towgs84=0,0,0,0,0,0,0
+units=us-ft +no_defs "
```

Expanding the SRS Table

The spatial_ref_sys table omits several continental and global map projections that are not defined by EPSG. Definitions and PostGIS Insert Statements for many SRS are available at Spatial Reference: `http://spatialreference.org/`



Geometry Columns

geometry_columns is a view stored in the public schema, use it to see the spatial attributes of features in the database.

```
SELECT *
FROM geometry_columns;
```

	f_table_catalog character varying(256)	f_table_schema character varying(256)	f_table_name character varying(256)	f_geometry_column character varying(256)	coord_dimension integer	srid integer	type character varying(30)
1	gep664	nyc	census_tracts	geometry	2	2263	MULTIPOLYGON
2	gep664	nyc	subway_complexes	geometry	2	2263	POINT
3	gep664	nyc	subway_stations	geometry	2	2263	POINT

There are separate views for geography_columns and raster_columns.

Transforming SRS

On the fly transformation from geometry in NY State Plane Long Island (ft-us) to simple NAD 83 long / lat

```
SELECT stop_id, stop_name,
ST_AsText(geometry) AS coord,
ST_AsText(ST_Transform(geometry,4269)) AS longlat
FROM nyc.subway_stations;
```

	stop_id character varying(3)	stop_name character varying(38)	coord text	longlat text
1	112	168 St - Washingt	POINT(1000814.97185065 245520.243242981)	POINT(-73.940133 40.8405559999997)
2	125	59 St - Columbus	POINT(989255.612340727 219170.38168695)	POINT(-73.981929 40.7682469999997)
3	127	Times Sq - 42 St	POINT(987714.521089981 214449.452072026)	POINT(-73.987495 40.7552899999997)
4	132	14 St	POINT(984194.298175778 208086.510156824)	POINT(-74.000201 40.7378259999997)
5	140	South Ferry Loop	POINT(980588.584176136 194819.684223018)	POINT(-74.013205 40.7014109999997)

Loading Spatial Data

Spatial files that have geometry (shapefiles):

- Use the shp2pgsql command line tool or shp2pgsql-gui (in MS Windows look under Start Menu - PostGIS Bundle - PostGIS Shapefile Loader).
- Alternatively - you can use the Database Manager in QGIS. It has a similar interface.
- Consider the loaded shapefile to be temporary. After loading, create a blank table to hold what you want, including geometry in the desired SRS. When inserting the data you can transform.

Loading Coordinate Data

Plain coordinate data (in csv or txt):

1. Can be loaded as a regular table using copy command into a blank table you create.
2. If the loaded table has the right structure and data, add a geometry column and populate it.
3. If it does not, consider the loaded table to be temporary. Create a blank table to hold what you want, including geometry in the desired SRS. When inserting the data you can transform.

Weather Station Example

Scenario 1 - Transform Existing Geometry Within Table

You have geometry, but not in the system you want. Alter the type of the column and transform the system. The weather stations are in NAD 83 - let's transform to NY State Plane.

```
ALTER TABLE nyc.weather_station
ALTER COLUMN geom TYPE geometry(point, 2263)
USING ST_Transform(geom, 2263);
```

Reminder:
EPSG 2263 is NAD83 NY State Plane Long Island (ft-us)
EPSG 4269 is NAD83

Weather Station Example

Scenario 2 - Transform & Insert Geometry in a New Table

You want to create a new table from a table that has geometry but it's not in the system you want. Transform it.

```
CREATE TABLE nyc.wstations_nysp (
station_id varchar(20) PRIMARY KEY,
station_name text,
elevation numeric(6,1),
lat numeric(9,6),
lon numeric(9,6),
geom geometry(POINT,2263));
```

```
INSERT INTO nyc.wstations_nysp (station_id, station_name,
   elevation, lat, lon, geom)
SELECT station_id, station_name, elevation, lat, lon,
ST_Transform(geom,2263)
FROM nyc.weather_station;
```

Weather Station Example

Scenario 3 - Create and Tranform Geometry Within Table

You have coordinates in one system and no geometry. Build the coordinates into geometry and transform them into the system you want.

```
ALTER TABLE nyc.weather_station
ADD COLUMN geom geometry(point, 2263);
```

```
UPDATE  nyc.weather_station
SET geom = ST_Transform(ST_SetSRID(ST_Point(lon,lat)
   ,4269),2263);
```

Weather Station Example

Scenario 4 - Build, Transform, & Insert Geometry in a New Table

You want to create a new table from a table that has no geometry and coordinates that are not in the system you want.

```
CREATE TABLE nyc.wstations_nysp (
station_id varchar(20) PRIMARY KEY,
station_name text,
elevation numeric(6,1),
lat numeric(9,6),
lon numeric(9,6),
geom geometry(POINT,2263));
```

```
INSERT INTO nyc.wstations_nysp (station_id, station_name,
   elevation, lat, lon, geom)
SELECT station_id, station_name, elevation, lat, lon,
ST_Transform(ST_SetSRID(ST_Point(lon,lat),4269),2263)
FROM nyc.weather_station;
```

Weather Station Example

Scenario 5 - Create a Spatial View

You can create a view to display data in a different system - it will even appear in GIS as a layer!
Transform:

```
CREATE VIEW nyc.wstat_nysp AS
SELECT station_id, station_name, elevation, lat, lon,
ST_Transform(geom,2263) AS newgeom
FROM nyc.weather_station;
```

Build and transform:

```
CREATE VIEW nyc.wstat_nysp AS
SELECT station_id, station_name, elevation, lat, lon,
ST_Transform(ST_SetSRID(ST_Point(lon,lat),4269),2263) AS
   newgeom
FROM nyc.weather_station;
```

Defining SRS

Use this if you have geometry that is missing a definition, or is defined incorrectly. Declare the desired SRID in both ALTER and USING. Note that the actual column name (geometry, geom, geom_nysp, etc) goes after COLUMN and ST_SetSRID while the TYPE is geometry.

```
ALTER TABLE nyc.subway_stations
ALTER COLUMN geometry TYPE geometry(POINT, 2263)
USING ST_SetSRID(geometry,2263);
```

Be careful - this is NOT the same as Transform. Here you are simply telling PostGIS what system the geometry is in. ST_Transform is what you use to change systems.

Geometry Potpourri

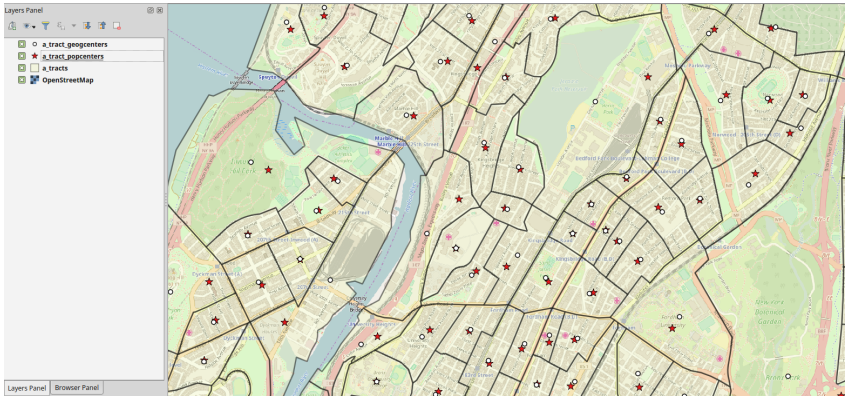
Geometry can also be displayed in a number of text formats using functions based on their names (i.e. ST_AsKML, and can be built from these formats using ST_GeomFromText and its variants (i.e. ST_GeomFromKML).

- WKB well-known binary
- WKT well-known text
- KML Keyhole Markup Language
- GML Geography Markup Language
- GeoJSON Geometry JavaScript Object Notation
- SVG Scalable Vector Graphics

Loading and Transformation Example

Load CSV, build geometry from coordinates and transform

Census tract population centroids represent the center of a population's distribution (red stars) rather than the geographic centroid (white circles).



Today's Topics

Spatial Reference Systems

SRS and PostGIS

Next Class

Midterm Prep

Due Next Class

The following are due at the beginning of our next class:

Assignment #7

Posted on the course website

Readings for Class #8

Listed in the syllabus, in the *PostGIS In Action* book

PostGIS in Action Readings

READ Chapters 9 & 11

But in these chapters you can skim or skip the following:

- 9.4.2 Using ST_Relate
- 11.4 Translating, scaling, and rotating geometries
- 11.5 Using geometry functions to manipulate and create geographies

Today's Topics

Spatial Reference Systems

SRS and PostGIS

Next Class

Midterm Prep

Midterm Quiz Part I

Midterm Quiz Takes Place at the beginning of Class #9 (two weeks from now) and is worth 10 points

Of these 7 terms, 5 will appear on the test. You will choose 3 to define in 4-6 complete sentences. You may not bring any notes (test is closed-book).

- Data type
- Geometry type
- Normalization
- Primary key
- Schema
- Spatial Reference System
- View

Midterm Quiz Part II

- You will be given print outs of 2 sample tables
- There will be 7 questions where you are asked to write a SQL statement
- Statements must follow the standard style guidelines
- Material covers the fundamentals from classes 2 & 3
- Except - there will be 1 question about adding geometry columns
- You will be given a SQL reference sheet that you can use during the test

The quiz begins promptly at 6pm. Please be on time.