Spatial Database Management
GEP 664 / GEP 380
Class #1: Introduction

Frank Donnelly

Dept of EEGS, Lehman College CUNY

Spring 2019

Today's Topics

Spatial Databases

Database that has been optimized to store spatial data and perform geographic functions.

Organization and storage: keep related geographic features and data tables together

Relational database: utilizes all the benefits of this structure

Multi-user: many people can read and write simultaneously

Automation and validation of tasks: connect to data with many tools, control over data quality

Work with big datasets: when desktop software and online tools fall flat

PostgreSQL & PostGIS

Relational Databases

An organized collection of data designed for efficiently answering questions and storing information.

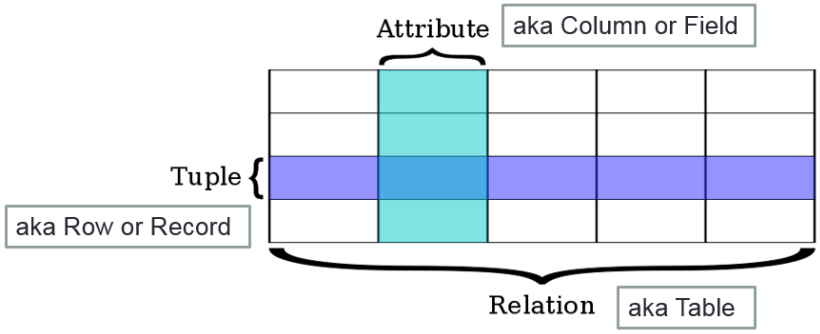


Image source: https://en.wikipedia.org/wiki/Relational_database

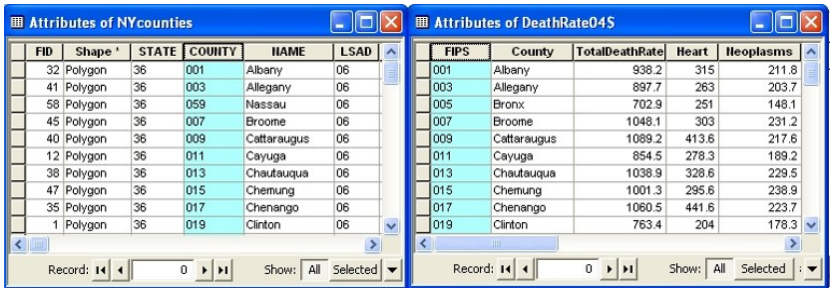
Data Structure

Values for specific attributes are drawn from an allowable set called a domain. Attributes are assigned data types, which limits the allowable values and operations that can be performed.

- Variable characters / Text (string)
- Integers (whole numbers)
- Reals / Floats (decimal numbers)
- Time and Date

Keys and Joins

This structure helps to insure the integrity of the data and makes it possible to relate values in one table to values in another using an attribute they hold in common: a unique ID code called a primary key.



SQL Data Manipulation

SQL is the language for creating and manipulating relational databases; originally based on relational algebra, it uses declarative commands in English.

```
SELECT county_name, pop AS population
FROM countypop
WHERE state='NY' AND pop > 50000
ORDER BY pop;
```

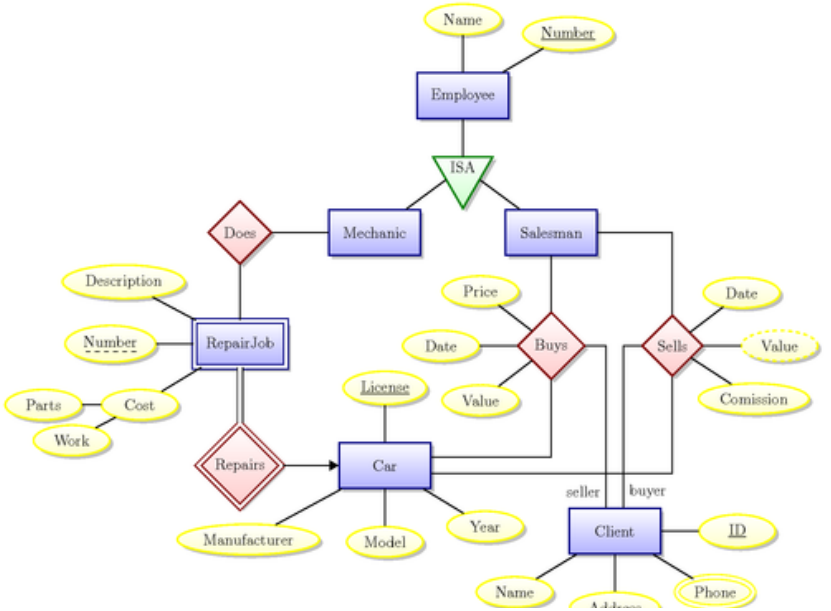
SQL Data Design

The structure of the relational database and the SQL language were designed to be independent of any specific hardware or software.

```
CREATE TABLE countypop (
geoid varchar(5)
CHECK (char_length(geoid) = 5),
state varchar(2),
county_name text,
pop integer,
CONSTRAINT gidkey PRIMARY KEY (geoid)
);
```

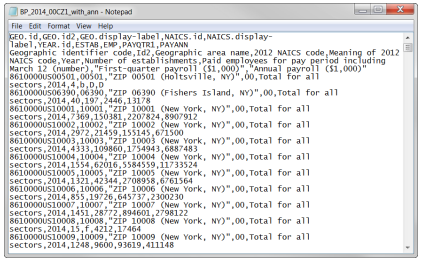
Database Design

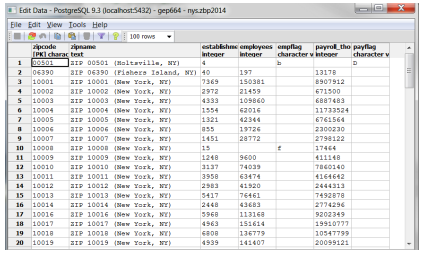
Image source: http://www.texample.net/tikz/examples/entity-relationship-diagram/



Data Processing

i.e. Extract, Transform, Load





Spatial Databases

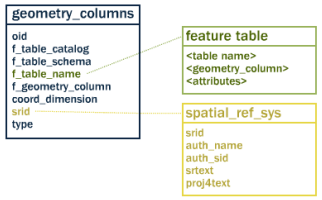
Geographic vector features stored as series of coordinates in a geometry data type. Geometry sub-types:

- POINT(0 0)
- LINESTRING(0 0,1 1,1 2)
- POLYGON((0 0,4 0,4 4,0 4,0 0),(1 1, 2 1, 2 2, 1 2,1 1))
- MULTIPOINT((0 0),(1 2))
- MULTILINESTRING((0 0,1 1,1 2),(2 3,3 2,5 4))
- MULTIPOLYGON(((0 0,4 0,4 4,0 4,0 0),(1 1,2 1,2 2,1 2,1 1)), ((-1 -1,-1 -2,-2 -2,-2 -1,-1 -1)))
- GEOMETRYCOLLECTION(POINT(2 3),LINESTRING(2 3,3 4))

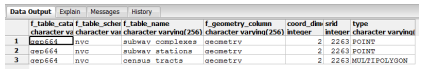
Spatial Reference Systems

Geometry columns are referenced with internal metadata tables that store coordinate systems.

Table Relationships




```
SELECT *
FROM geometry_columns;
```



	f_table_cata character va	f_table_schen character va	f_table_name character varying(256)	f_geometry_column character varying(256)	coord_dim integer	srid integer	type character varying(
1	gep664	nyc	subway complexes	geometry	2	2263	POINT
2	gep664	nyc	subway stations	geometry	2	2263	POINT
3	gep664	nyc	census tracts	geometry	2	2263	MULTIPOLYGON

Spatial Relationships & Analysis

When features have geometry they can be compared spatialy.

```
SELECT bname, stop_name, trains
FROM boroughs, stations
WHERE bname='Bronx' AND
ST_Within (stations.geom, boroughs.geom);
```

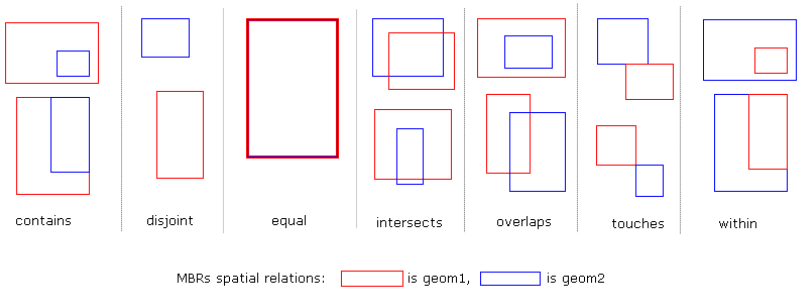
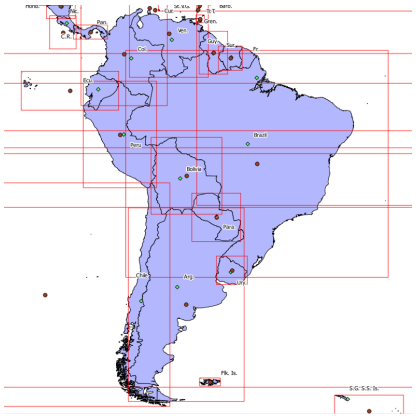


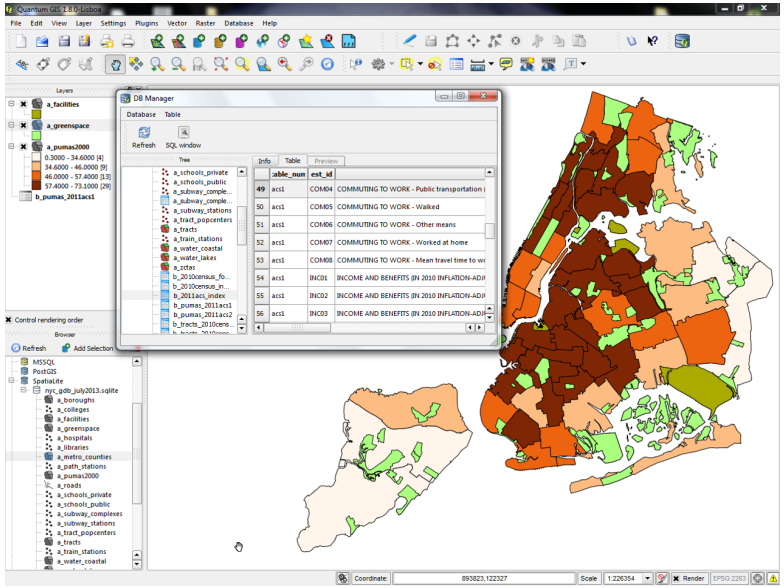
Image source: https://www.gaia-gis.it/spatialite-2.1/SpatiaLite-manual.html

Proximity Analysis

Adjacency, Buffers, Distance, Distance Within, Nearest Neighbor. Spatial relationships can be evaluated quickly using spatial indexes.



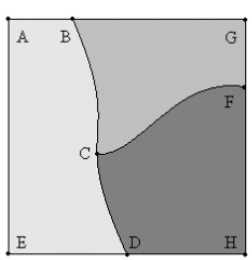
Spatial Data Organization & Desktop GIS



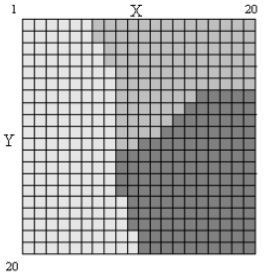
Database Management

- Roles and permissions for users
- Triggers and transactions for processes
- Fine tuning and performance
- Backup and restoration

Rasters



Vector image



Raster image

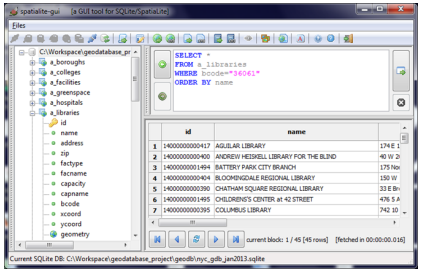
Vector		
Polygon ID	Coordinates	Soil Type
1	A,B,C,D,E	Chalk
2	B,C,F,G	Clay
3	C,F,H,D	Gravel

Raster	
Grid Ref.	Item
x=1, y=1	Chalk
X=2, y=1	Chalk
X=3, y= 1	Chalk
X=4 ... etc.	...
X=20, y=20	Gravel

Image from http://www.arts-humanities.net/wiki/gis_geographic_information_system_archaeology

Other Database Formats

Spatialite for desktop, ArcGIS formats



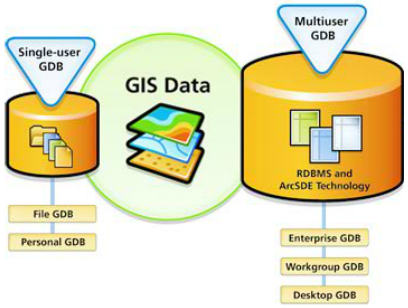


Image source: http://www.esriuk.com/software/arcgis/geodatabase

Programming / Scripting Languages

Python and PostgreSQL

Course Structure

Classes 1 through 5
Essentials of Relational Databases (Jan - Feb)

Classes 6 through 10
Essentials of Spatial Databases (Mar - Apr)

Spring Break
Apr 23

Classes 11 through 13
Special Topics (Apr - May)

Class 14 and Final
Final Projects (May)

Order of Operations

For each class:

1. Hand in assignments
2. Lecture and discussion
3. Exercises
4. Break
5. Lecture and discussion
6. Exercises
7. Return previous assignments and discuss
8. Lab time to begin next assignments

My Background

Geography & Library & Information Science

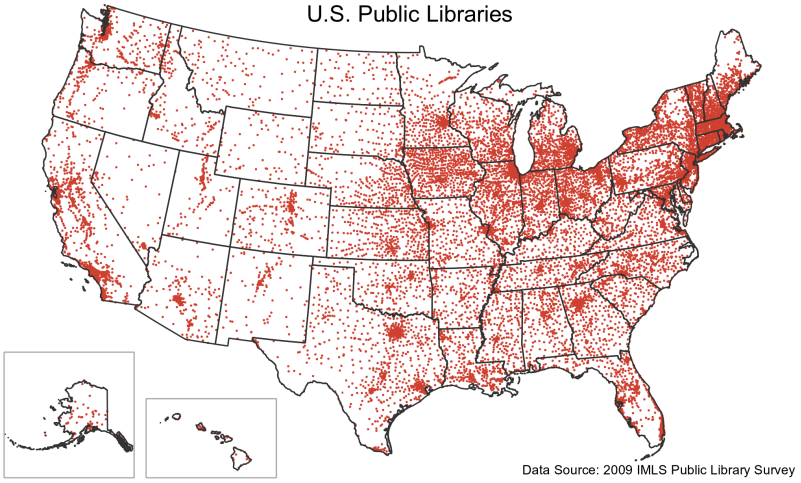


Image source: https://atcoordinates.info/tag/public-libraries/

Today's Topics

The Syllabus

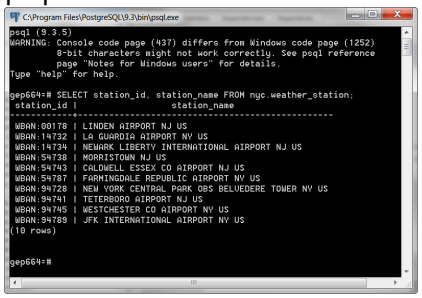
Course website: `https://spatdb.commons.gc.cuny.edu/`

Today's Topics

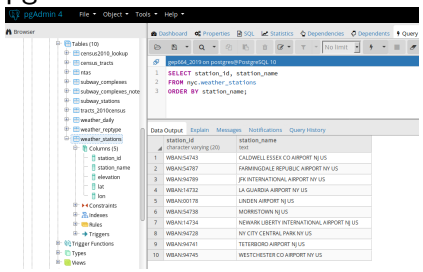
Interfaces

Databases exist as discrete objects outside of a specific interface. You can interact with a CLI or GUI.

psql



pgAdmin 4



Architecture

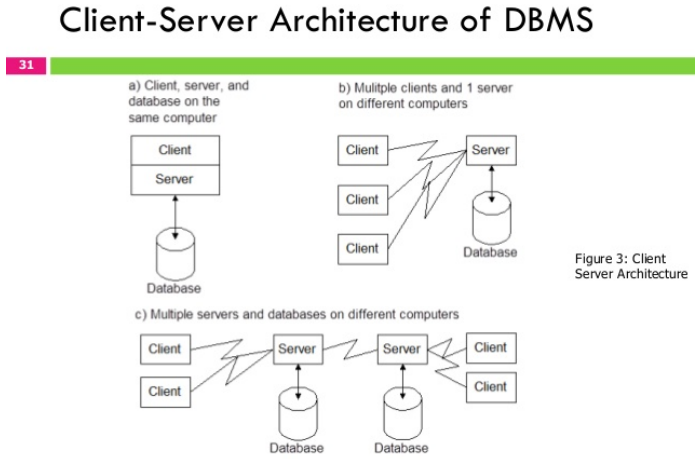


Figure 3: Client Server Architecture

Image source:
`http://www.slideshare.net/b15ku7/chapter-1-fundamentals-of-database-management-system`

Today's Topics

Due Next Class

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

Assignment #1

Posted on the course website (under Assignments)

Readings for Class #2

Listed in the syllabus, in the *Practical SQL* book

Note: There is overlap in course content and readings for classes 2 & 3