

**11.520: A Workshop on Geographic Information Systems**

**11.188: Urban Planning and Social Science Laboratory**

## **Lecture 4: Relational Databases & GIS Data Models**

**September 28, 2005, Joseph Ferreira, Jr.**

**(including contributions from Visting Prof. Zhong-Ren Peng who taught the class Fall, 2003)**

---

### GIS Data Models

- Computer Aided Design (CAD), graphical, and image GIS data models;
- Raster data model;
- Vector data model;
- Linear Reference data model;
- Network data model;
- TIN data model.

The contents of the GIS data model section of today's lecture is derived from Longley, Goodchild, Maguire and Rhind, *Geographic Information Systems and Science*, 2001, as organized by Prof. Zhong-Ren Peng for 11.520 in Fall 2003.

### CAD data models

- In CAD, real-world entities are represented symbolically as points, lines and polygons.
- CAD data model is different from GIS data models:
  - CAD models uses local drawing coordinates rather than real-world coordinates.
  - Individual objects in CAD do not have unique identifiers and attributes.
  - CAD data model does not store details of relationships (e.g., topology) between objects.

## Computer Cartography

- The purpose of computer cartography is to automatically reproduce paper maps.
- All paper map entities are stored as points, lines and polygons, with annotations used for placement.
- Only limited attribute data are associated with entities (enough for symbology).
- Relationships among entities are not stored.

## Image data model

- Use images (photos, aerial photos and satellite images) to represent real world entities.
- Working with annotated pictures or real world entities.

- Images need rectification and registration to be integrated with other georeference data. (e.g., GeoTiff vs. Tiff image format)

## Raster data model

- The raster data model uses an array of cells, or pixels, to represent real-world space and then encodes the grid cells based on objects in the cell.
- The cells can hold any attribute values based on different encoding schemes.
- Raster data are usually stored as an array of grid values, with metadata held in a file header.
- Typical metadata includes geographic coordinate of the upper-left corner of the grid, the cell size, and the number of row and column elements.

An example of a raster dataset with integer and floating point grid cell values (e.g., layer1 = soil type at center of grid cell, layer2 = cell average groundwater depth in meters)

Row	Column	Layer1_value	Layer2_value
1	1	1	2.0090
1	2	0	12.665
1	3	2	1.2211
2	1	1	3.3566
...	...		

## Vector Data Model

- Vector data model describes the *boundaries* of real-world objects using 2D geometric types: point, line, or polygon and a real-world coordinate system
- Simple features model the basic object geometry
- Topologic features model relationships among objects

## Simple Features

- Geographic entities encoded using the vector data model are called features.
- Features are vector objects of type point, line or polygon.
- Lines and polygons can overlap.
- There is no stored relationship between any objects.
- Simple feature data structure is sometimes called spaghetti.
- See textbook Longley et al. (2001) pp. 190.

## Advantages and drawbacks of simple features

- Easy to create and store.
- Easy to retrieve and render on screen.

- Inefficient to store, boundaries of two adjacent polygons - shared boundaries need to be stored twice.
- Inflexible in dissolving common boundaries when joining two zones or editing geometry to move common boundary points.
- May result in gaps (slivers) or overlaps of polygons.
- Many operations cannot be performed due to the lack of connectivity relationships in the data structure: e.g., finding the shortest path through a road network.

## Topologic features

- Topologic features are simple features structured using topologic rules.
- Topology is the science and mathematics of encoding shape and spatial relationships.
- Topology can be used to validate the geometry of vector entities (e.g., polygons that aren't 'closed').
- Topology can be used for certain operations such as network tracing and tests for polygon adjacency.

## Topologic structure – Line

- A line is defined as a directed sequence of points from a starting node to an ending node.

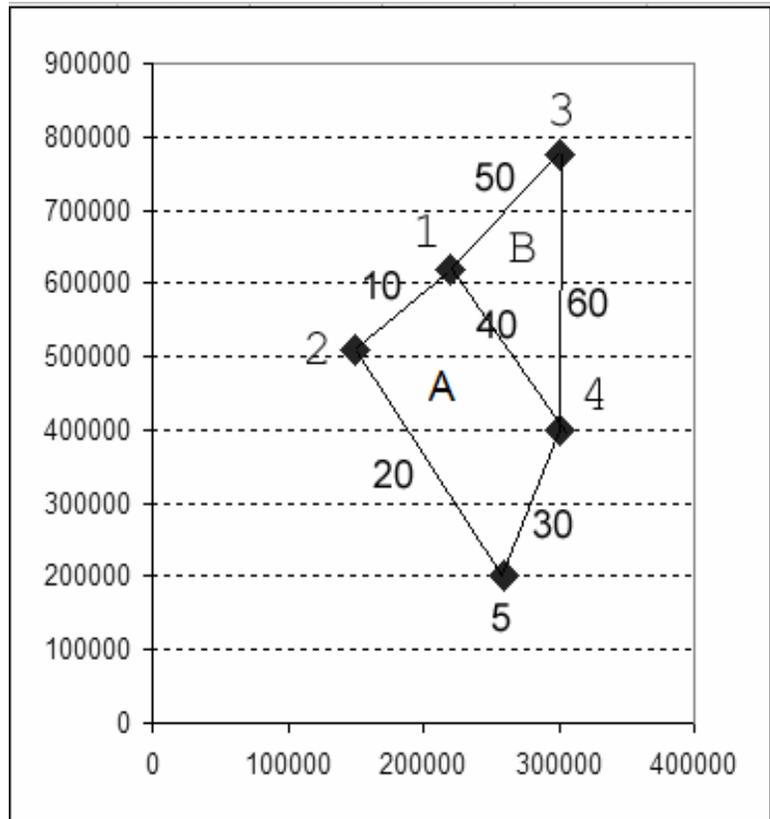
- Points (vertices or line end nodes) that fall within a minimum tolerance of each other are snapped together.
- New nodes are created wherever two lines intersect.
- Hence, the name “spaghetti with meatballs”.

## Topologic structure – polygon

- Polygons are defined as a sequence of lines that enclose an area.
- Points are stored once in a point list; lines are stored as a directed sequence of point IDs; polygons are stored as a directed sequence of line IDs. Moving a point, will automatically move all the lines and polygons that utilize the point.
- <![endif]> Planar enforcement: all the space on a map must be filled and any point must fall in one polygon only. That is, polygons must not overlap.

PointID	X-value	Y-value
1	200101.22	620112.3
2	150000.00	510000.4
3	300100.10	777000.1
4	300000.00	400000.00
5	260000.00	200000.00
LineID	FromNode	ToNode
10	2	1
20	2	5

30	5	4
40	4	1
50	1	3
60	4	3
PolygonID	LineID sequence	
A	20, 10, 40, 30	
B	50, 60, 40	



## Contiguity (adjacency) relationship

- Each line must have a direction to define contiguity (adjacency) relationship.
- Defined by a list of polygons on the left- and right-hand side of each line.

- An example is given at the textbook Longley et al (2001) pp. 191, Figure 9.9.

## Geo-Relation model

- The Geo-relation model refers to software that implements the vector topologic feature data model.
- Store the geometry and associated topologic information in one set of files (with object IDs).
- Store associated attribute information in relational database management system (with object IDs as foreign keys into geometry and topology data).
- The GIS software maintains the linkage between geometry, topology and attribute information.

## Network data model

- Networks are modeled as points (nodes) and lines.
- Network topology defines how lines connect with each other at nodes.
- There are also rules to define how flows can move through a network, like one-way street, sewerage flow, etc.
- The rate of flow is modeled as impedance on the nodes and lines, such as turn signals and speed limits.

## Linear Referencing

- Store the location of geographic entities (called events) as a distance along a network (a route system) from a point of origin.
- Offsets are used to store information about the distance from the centerline.
- Dynamic segmentation is a special type of linear referencing. It assign event data to the network segments dynamically.

## Triangulated Irregular Network (TIN) data model

- TINs are used to create and represent surfaces in GIS.
- The TIN structure represents a surface as contiguous non-overlapping triangular elements.
- TIN is created from a set of sample points with x, y, and z coordinate values.
- An example of TIN data sturcture is given at the textbook Longley et al (2001) pp. 195, figures 9.12 and 9.13.

## Data Model Examples from Lab Exercises

- Vector: *objects* as points, lines, polygons
  - ArcGIS data files:
    - Coverages: old Arc/Info a directory per layer, plus INFO files

- Shapefiles: .shp, .shx, .dbf files (and possibly others)
- Spatial Database Engine (SDE):  
retrieved dynamically from a database server
- Lab exercises 1 and 2:
  - Cambridge blockgroups (polygons in shapefile)
    - Look at cambbgrp.dbf from MS-Access (beware of sort order for matching geometry)
  - Sales89 (points in shapefile)
    - Can't view sales89\_shape.dbf in MS-Access because name is too long for Access!
    - Lat/lon columns in a data table can be used by ArcMAP to create point geometry and build a shapefile (the sales89 shapefile was built by geocoding street address to lat/lon and building point layer from the lat/lon)
- Raster: *space* as 2D or 3D grid cells
  - regular grid on top of spatial features (instead of encoding boundary)
  - pixel brightness in orthophoto of Boston
  - Raster example: orthophotos, [scanned maps](#), grids
  - Lab exercise 2:
    - MITOrthoTool brings in orthophoto snippets as TIFF images

- TIFF image doesn't know 'where' it is but ArcMAP script has already figured that out



Boston/Cambridge Streets superimposed on orthophoto. Zoomed-in view shows raster nature of the ortho.

- Complications
  - islands, lakes, overpasses
  - share edges?, move links when you move points?
  - ambiguity: summer/winter wetland boundaries

- scale/level-of-detail, generalization, conflation, slivers

## Relational Databases

- One 'flat file' data table for each map layer is too simple a data model
  - From lab exercise #3: because cambbgrp was a read-only shapefile, we had to build a new table with percent-with-high-school-education computations and then join it to the original **cambbgrp** table
  - From lab exercise #3: some houses in **sales89** may have sold more than once! But, there's only one dot for the location of the house
  - Keep tabular model for handling textual data but allow many tables that can be related to one another via common columns (attributes)
    - Allows us to extend attribute tables by adding additional data for each blockgroup
    - Handles one-to-many situations such as multiple sales for the same house
    - Provides many other powerful mix-and-match capabilities
- The relational database model and the structured query language (SQL) for joining tables and specifying queries is the lingua franca of distributed database operations on public (internet) and private networks (bank ATM transactions, airline reservations)
- Continue in Monday's lab and next Wednesday's lecture with more on relational database management

and queries - especially, handling one-to-many relationships via ArcMap 'summarize' tools, and MS-Access queries.

---

*Last modified 28 September 2005. [jf]*