GSB/RF
November 26, 1999

Project 3 — A Rudimentary GIS System

Introduction

The goal of this project is to implement a program for the visualization of geographical data. The intent is solve a realistic problem on a real-world set of data.

The data will be the TIGER/Line® Files 1997 from the US Bureau of Census. The files contain a large collection of annotated points, lines and polygons, which together form a detailed map of the United States.

The basic task is to build an external memory data structure holding a suitable subset of these data, and to make a program which take a specification of a rectangle – a *window* – and graphically display the part of data contained in it.

The project is intentionally left quite open. The version described below is a required minimum, but students are free to make a more advanced solution. Some ideas for possible enhancements are mentioned later.

The TIGER/Line Data

The TIGER/Line data set is produced by the US Census Bureau (`www.census.gov`). The following description of TIGER/Line is taken from the Bureau's web page for the TIGER/Line data set (`www.census.gov/geo/www/tiger`).

> The TIGER/Line files are a digital database of geographic features, such as roads, railroads, rivers, lakes, political boundaries, census statistical boundaries, etc. covering the entire United States. The data base contains information about these features such as their location in latitude and longitude, the name, the type of feature, address ranges for most streets, the geographic relationship to other features, and other related information. They are the public product created from the Census Bureau's TIGER (Topologically Integrated Geographic Encoding and Referencing) data base of geographic information. TIGER was developed at the Census Bureau to support the mapping and related geographic activities required by the decennial census and sample survey programs.
>
> These files are not graphic images of maps, but rather digital data describing geographic features. To make use of these data, a user must have mapping or Geographic Information System (GIS) software that can import TIGER/Line data. The Census Bureau does not provide these data in any vendor-specific format. With the appropriate software a user can produce maps ranging in detail from a neighborhood street map to a map of the United States. To date, many local governments have used the TIGER data in applications requiring digital street maps. Software companies have created products for the personal computer that allow consumers to produce their own detailed maps. There are many other possibilities.

The TIGER/Line 1997 data set is available from the web site of the University of California, Library at Berkeley (`sunsite.berkeley.edu/GovData/info/tiger.html`). The TIGER/Line 1997 data set consists of six CDs each containing approximately 600 Mb of compressed data, which expands to approximately 5 Gb data for each CD. The data for New York state (NY) on Disc 1 (`/tiger/36_ny/`) is 86 Mb and uncompresses to 729 Mb, describing approximately four million lines.

The documentation for the data files can be found on every CD in the file `/techdoc/tiger97d.pdf` (273 pages).

The Minimal Requirements

The following are the minimum requirements of a solution:

- The data should consist of the lines for the roads, the railroads and the waterways (Feature Class A, B and H, see Section 3 of `tiger97d.pdf`). Only the actual lines are mandatory, i.e. no annotation needs to be shown, and no polygons defined by the lines need to be considered as separate entities. The necessary information for this is contained in the records of type 1 and 2. The complete set of points for a polygon line (a *chain* in TIGER terms) is obtained by merging the information in the files for the two types of records. For this, they files should first be sorted according to the TLID key (TIGER/Line Identification Number).
- The data should be stored in a static external memory R-tree. The R-tree should be build in a bottom-up fashion on the set of lines sorted according to the position of the centers of their bounding boxes along the Z-curve. This is equivalent to sorting on a key obtained by interleaving the bits of the x- and y coordinates of the centers.
- A query program should be made which takes two points and displays the lines or part of lines contained in the rectangle spanned by these two points. The program does not have to be interactive.
- The method of generating graphical output may be chosen rather freely. A graphical library may be used for direct on-screen output, or Postscript or LaTeX-files may be generated. Other methods may be approved on request.
- The amount of data should be such that the data structure storing them takes up at least 0.5 Gb of disk space. The amount should be adjusted by choosing a proper number of (neighboring) states. The state New York should be included.
- The program should be implemented in C or C++.

Enhancements

Among the possible improvements of the minimal solution are:

- Store more types of data (e.g. landmark and polygon features).
- Show annotations (names, etc.).

- Make an interactive query program, with navigational features.
- Make a zoom facility which adjusts the level of detail according to the size of the window. This may be implemented by grouping the subclasses of the feature classes according to some judgment about what level of detail they belong to. Only the classes for the appropriate level should then be shown.
- Substitute the R-tree (which gives no guarantee on the worst case query time) by more advanced data structures, e.g. the ad hoc externalization of extended segment trees described in a lecture. The problem of finding line segments intersecting a given rectangle can be solved by such a tree in combination with a one-dimensional range tree (i.e. a B-tree), as described in the lecture (details can be found in Chapter 10 of the book from the computational geometry course, *Computational Geometry: Algorithms and Applications*, by de Berg, van Kreveld, Overmars and Scharzkopf, Springer, 1997).

Many design choices are left open in this project. Some of these may be resolved by experimentation. Such experimentation is encouraged (and so is its documentation in the report).

Due date

A report documenting the work carried out and the solution obtained should be handed in by Saturday January 15, 2000. Extension may be negotiated if a group intents to implement an enhanced solution.