

Importing Abstract Spatial Data into the SAND Database System*

Hanan Samet
Frantisek Brabec
Jagan Sankaranarayanan
Computer Science Department
Center for Automation Research
Institute for Advanced Computer Studies
University of Maryland
College Park, Maryland 20742
www.cs.umd.edu/~hjs

E-mail: {hjs,brabec,jagan}@cs.umd.edu

1 Introduction

The World Wide Web has opened ways to operate large applications by users with simple client platforms located half way around the world by connecting servers around the world together. We present a system that makes a centrally stored spatial database available to off-site users. Regardless of the specific platform available to a user, all they need to do is simply establish a link between their client and the server. Unlike well-known web-based services such as MapQuest [2] that rely on computational power of the server, our system distributes the workload among the client and the server in such a manner that the user will observe the system as being interactive, with minimal delay between the user action and appropriate response for most types of operations.

The SAND Internet Browser, the client part of our solution¹, provides a graphical user interface for viewing and querying data stored on the central server. One of our design goals was to minimize specific hardware and software requirements for the client component. Our solution was to implement the SAND Internet Browser in Java, this provides maximum portability across multiple hardware platforms while not requiring anything beyond a common networked computer, i.e., an Internet-connected computer with a Java-enabled web browser.

The SAND Internet Browser is continuously being improved and new functionality has been added since its earlier presentations such as [3, 4, 6, 7]. This paper accompanies a demo which introduces some of the features that have been recently added or are currently being researched and will be available in the near future.

2 Importing Data

One of the features our users and supporters have been asking about most is allowing data exchange with other systems

*This work was supported in part by the National Science Foundation under Grants EIA-99-00268 and EIA-00-91474.

¹This client can be downloaded and launched against our sample spatial server by visiting <http://www.cs.umd.edu/~brabec/sandjava/index.html>.

that support spatial data. Note however, that such systems may not refer to spatial information in terms of coordinates; instead, they may be using textual information such as street addresses or intersections. We have investigated this matter and as a result of our research, SAND can now convert a number of standard formats by directly importing them into the internal SAND format. The SAND Internet browser includes a spreadsheet-like interface that allows data to be manually edited before imported into the database. This approach is suitable for semi-structured data like Excel, HTML and GML that may contain inaccuracies or need human intervention in order for the conversion process to execute successfully. The ultimate goal of our efforts is to allow SAND to import data that provide information meaningful to human intellect yet that are not expressed in the exact format computers can readily understand. Our examples below show how spatial data valid from human perspective can be processed into a specific format supported directly by spatial DBMSs. We show that importing such abstract entities to a computer system may not be a trivial task.

2.1 Postal Data

A postal address represents a unique geographical location. A postal address consists of a door number, a street name, a city name and postal/zip codes. It is one of the common methods that humans use to locate a point in a road network. Large quantities of data with address information exist on the Internet. Unfortunately, these datasets where information is expressed in textual postal address format are difficult to query using their spatial properties (e.g., distance or nearest neighbor queries). The conversion of postal addresses to latitude/longitude pairs is called geocoding. To achieve geocoding functionality for locations in the U.S., we converted the Tiger [1] Shapefiles of all counties using the built-in routines on the SAND Browser. Next, a spatial index was created on these datasets. The naive way to perform geocoding would be to perform textual searches directly on the addresses where there is an index correlating each entity's textual representation with the location in space through which it passes.

Clearly, such a method would not make use of the spatial

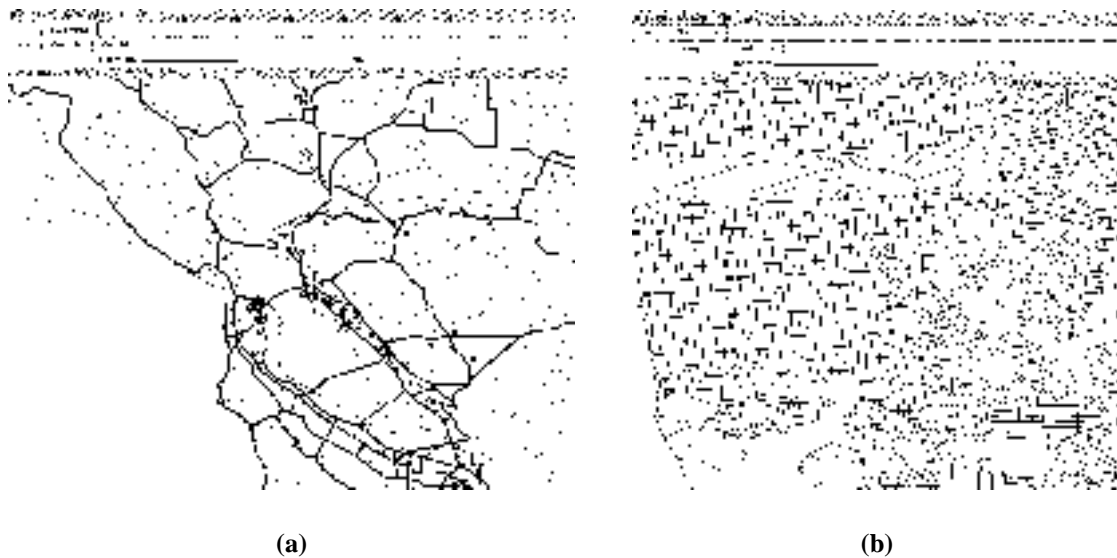


Figure 1. SAND Internet Browser User Interface — (a) HUD Projects sites after the textual addresses were geocoded. (b) Locations of bus stops on a San Francisco road map.

nature of the information. Instead, we devised a geocoding routine that iteratively shrinks the search area until it is reduced to the smallest possible bounding box. For example, given an address we first find the smallest bounding box that encloses the corresponding state. Within this bounding box, we search for the county name thereby reducing the bounding box further. We continue reducing the bounding boxes as long as possible. At that point, the actual textual search is performed on the address within the smallest bounding box that encloses all streets that lie within the same zip code. Once we match addresses to the corresponding locations, the resulting dataset can be visualized and queried in spatial context (Figure 1a).

2.2 Locating road intersections

Working with bus schedules presents another scenario where the need for conversion of textual information into coordinates arises. Bus schedules represent routes that buses take and each bus schedule is often represented in terms of road intersections (Figure 1b). This information in itself is insufficient to map the route into geographical location. This class of problem is a more difficult than the previous example. A bus stop is represented as a pair of street names where the streets involved are known to intersect at some point.

A naive approach could involve searching for all pairs of intersecting streets but this is too computationally difficult. A better approach will use the spatial nature of the information available. An example of such an approach assigns Morton codes [5] to each street segment on the list, then sorts the list in the order of their Morton codes. The intersection problem can then be solved more efficiently by simply comparing the street Morton codes. If the codes are the same, the streets intersect. This process makes the problem much easier to solve.

2.3 Conclusion

We have shown some of the advances we have made in our research on spatial databases, client-server architecture, and the SAND Browser specifically. Our team's goal is to make this application powerful as well as intuitive enough so that this technology can be utilized by 3rd party institutions involved in working with spatial data. Examples of such applications will be demonstrated.

References

- [1] Census 2000 tiger/line data provided by U.S. bureau of the census. url:http://www.esri.com/data/download/census2000_-tigerline.
- [2] Mapquest is a leading provider of interactive location-based services url: <http://www.mapquest.com>.
- [3] C. Esperança and H. Samet. Spatial database programming using SAND. In M. J. Kraak and M. Molenaar, editors, *Proceedings of the 7th International Symposium on Spatial Data Handling*, volume 2, pages A29–A42, Delft, The Netherlands, Aug. 1996. International Geographical Union Commission on Geographic Information Systems, Association for Geographical Information.
- [4] C. Esperança and H. Samet. Experience with SAND/TCL: A scripting tool for spatial databases. *Journal of Visual Languages and Computing*, 13(2):229–255, Apr. 2002.
- [5] G. M. Morton. A computer oriented geodetic data base and a new technique in file sequencing. , IBM Ltd., Ottawa, Canada, 1966.
- [6] H. Samet, H. Alborzi, F. Brabec, C. Esperança, G. R. Hjaltason, F. Morgan, and E. Tanin. Use of the SAND spatial browser for digital government applications. *Commun. ACM*, 46(1):63–66, Jan. 2003.
- [7] H. Samet, F. Brabec, and G. R. Hjaltason. Interfacing the SAND spatial browser with FedStats data. In *Proceedings of the dg.o 2001 Conference: Connecting Government and the People Electronically*, pages 41–47, Redondo Beach, CA, May 2001.