Visualization Blackboard

Visualizing Large Telecommunication Data Sets

Eleftherios E.
Koutsofios,
Stephen C.
North, and
Daniel A. Keim

*AT&T
Laboratories*

Global telecommunication services create an enormous volume of real-time data. Long-distance voice networks, for example, can complete more than 250 million calls a day; wide-area data networks can support many hundreds of thousands of virtual circuits and millions of Internet protocol (IP) flows and Web-server sessions.

Unlike terabyte databases, which typically contain images or multimedia streams, telecommunication databases mainly contain numerous small records describing transactions and network status events. The data processing involved therefore differs markedly, both in the number of records and the data items interpreted.

To efficiently configure and operate these networks, as well as manage performance and reliability for the user, these vast data sets must be understandable. Increasingly, visualization proves key to achieving this goal.

AT&T Infolab is an interdisciplinary project created in 1996 to explore how software, data management and analysis, and visualization can combine to attack information problems involving large-scale networks. The data Infolab collects daily reaches tens of gigabytes.

The Infolab project Swift-3D uses interactive 3D maps with statistical widgets, topology diagrams, and pixel-oriented displays to abstract network data and let users interact with it. We have implemented a full-scale Swift-3D prototype, which generated the examples we present here.

Swift-3D

We developed Swift-3D to address two needs in network-element management: increased level of abstraction and improved real-time analysis capabilities. Our goal was to help decision makers react to changing conditions within a matter of minutes. We thus focused Swift-3D on visualizing circuit-switched call detail information.

To be useful, a system for visually exploring call detail records must scale well and avoid instance-specific processing, making it flexible enough for experimentation in back-end queries and user interface. Although initially aimed at voice telephony, our data mining and visualization model can generalize or scale to other large-scale networks and services.

Swift-3D has three modules: data collector, aggregator, and visualization interface. The modules communicate using a self-describing data-independent binary format consisting of

- a header that defines record size, type, and data context; and
- a record sequence.

We created our own data file format after our initial studies found that commercial databases either couldn't handle such large volumes or imposed inordinate overhead and administrative complexity for our purposes.

Swift-3D serves for batch playback, which is faster than real time, or for real-time applications in network management, where incremental state changes transmit through its data paths. We built Swift-3D's underlying query language using C-like expressions—such as arithmetic, logical, and statistical operations and groupings—that are compiled, loaded, and executed on-the-fly for efficiency. Because Swift-3D runs on data sets potentially much larger than physical memory, its visualization module explicitly controls paging via memory-mapped files.

Swift-3D uses multiple linked 2D and 3D views, including geographic maps and statistical displays and controls. Most display objects also support input, such as "drill down when an interesting pattern is found."

Figure 1 shows a screenshot of the Swift-3D system. The upper left window shows US traffic volume over 10-minute intervals for several services and their aggregate. The upper right window shows a subset of the database for display. The large window shows a 3D display with a histogram bar for each location. The mapping between data and display is customized in an auxiliary geometry file of points, lines, polygons, and coloring. Users can define new maps to highlight interesting properties or encode multiple levels of detail, such as from state and county outlines down to local telephone exchange or nine-digit postal zip boundaries. They can then relate the new details to the external values displayed. Swift-3D generates the 3D graphics using Silicon Graphics Performer and other libraries and tools.

In Swift-3D, users can examine data from different viewpoints, zoom in on interesting locations, and select intervals for viewing and animating time-series data. A drag-and-drop interface in the visualization window lets users drill down to get details, explore context, and take
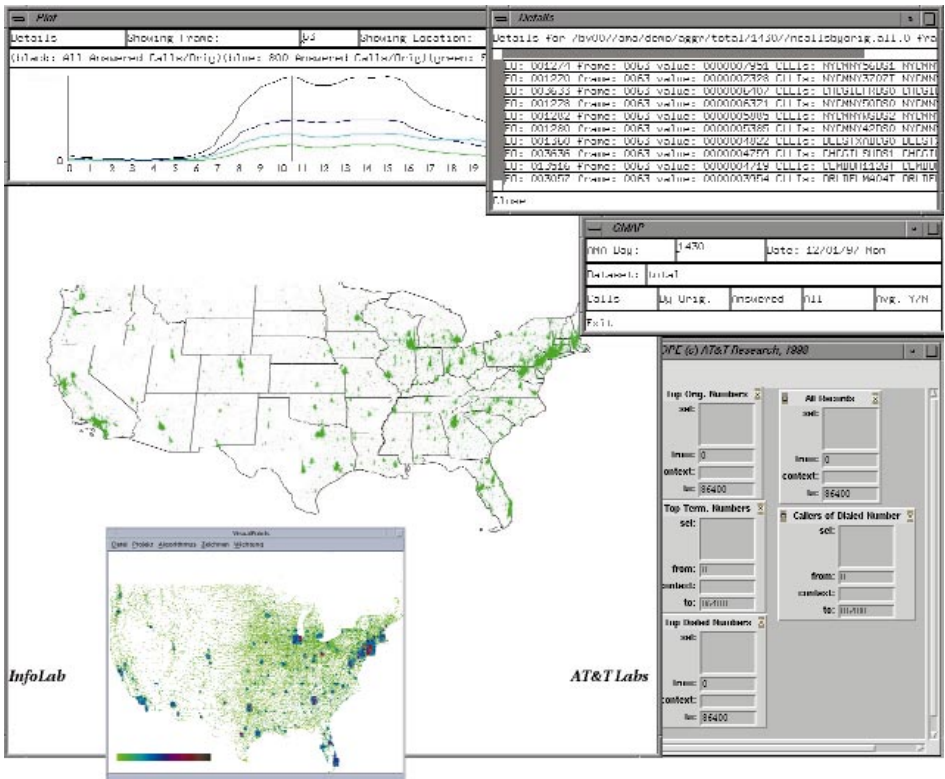
1 An overview of the Swift-3D display. This example shows US traffic volume over 10-minute intervals for several services and their aggregate (upper left); a subset of the database (upper right); and, in the large window, a 3D display with a histogram bar for each location. Users can also open a 2D overview (lower left), which shows data for each location in a single pixel color.

action if necessary. This method intuitively converts spatial information into detailed information, such as the most frequent originating or terminating numbers.

As the lower left of Figure 1 shows, users can also open a 2D overview window that shows data for each location in a single pixel color. We generated the 2D displays using an extension of the VisualPoints system,[1] arranging data points on the screen so that those with similar coordinates—typically mapped to the same position—are represented by different colored pixels. High call volumes are mapped to dark colors and low call volumes to light colors.

Network analysis

Figure 2 shows a random estimate of telephone-call volume on a typical weekday in the US. The view focuses on the activity at geographic network endpoints, from the west to east coasts. When run interactively, the display shows network usage in relation to time: changing as users wake up, commute to work, take lunch breaks, and return home to make personal calls and log in to Internet services. Some trends aren't so predictable and invite further investigation. For example, even after business hours end, some eastern metropolitan areas continue to log relatively high activity volume.
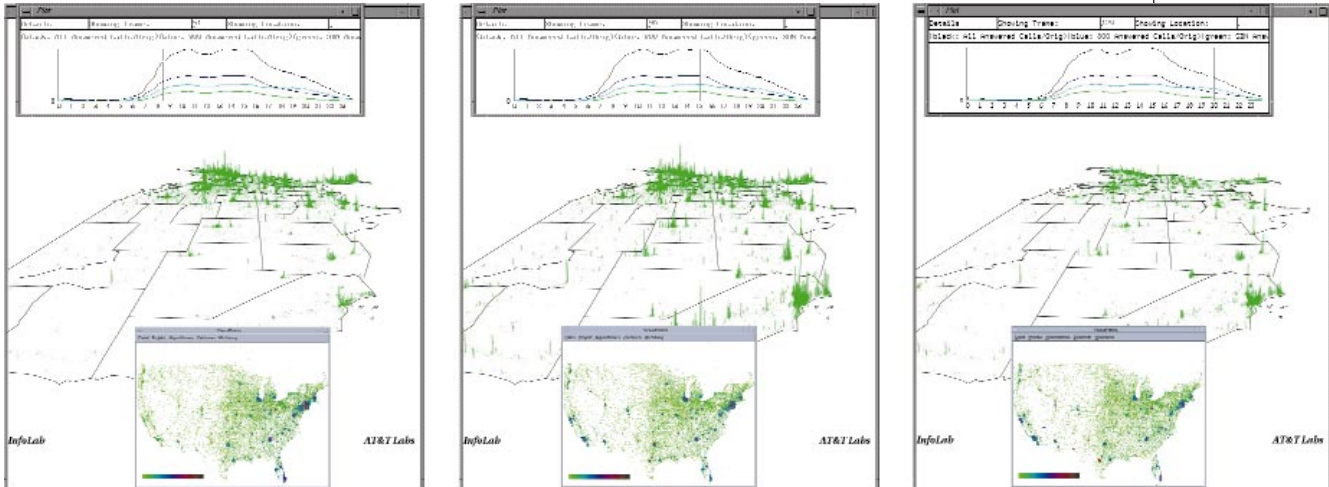
2 Network traffic time-series animation. When activated, the display of call volume shows fluctuations in network usage over time. In each frame, the upper window show the volume of specific voice services, such as residential, long distance, and 800-number calls.

3 Inspection of a network event's impact on customers of a toll-free call center. The display shows the effects of a certain event on geographic locations and originating and terminating numbers.

4 Virtual network configuration created by a customer. The display was made by joining network topology and customer-configuration databases.

A network endpoint view also suits exploring second-order effects or network events. For example, users might look at call volumes normalized by population, or calls that went unanswered or otherwise failed to complete due to a specific hardware or service outage.

Applications

Swift-3D supports interactive queries on a database of more than 250 million records per day. It can thus visualize network activity and events for many different businesses and services, as the following examples show.

For example, companies that operate large toll-free call centers must monitor customers' quality of service. Figure 3 shows the volume of unanswered toll-free calls during a major network problem. To explore details about the problem's impact on a certain customer, the Swift-3D user can select a data set to display, isolate a specific region where numerous events indicate incomplete calls, and use drag-and-drop tools to find out who called into those locations, sorted by count. This lets network managers assess both the cause and impact of a network or customer equipment problem and suggest steps to reduce the disruption.

In one case, managers discovered that calls to a radio station dropped at a high rate not because of a network error, but because the customer had only a single line and received thousands of calls per hour.

Swift-3D can also help users configure virtual networks. In Figure 4, for example, a large corporate customer configured a virtual network by joining network topology and customer configuration databases. Although many user-configured networks exhibit a similar star-like structure, others are mesh-like, possibly reflecting a less centralized application architecture. Further research on ways to reorganize the layout of virtual networks on physical facilities might improve network performance and reliability.

Swift-3D also proves useful for market analysis, which can help managers better understand market-segment behavior and thus improve their marketing strategies. This often involves comparing multiple data sets that vary by time, market segment, service, or provider.

For example, Figure 5 shows the relationship between voice long-distance service and Internet access. The display shows

- differences in the use of services by time of day,
- local telephone calling areas (where Internet access does not involve a per-minute connection charge), and
- the local calling areas of a competitor.
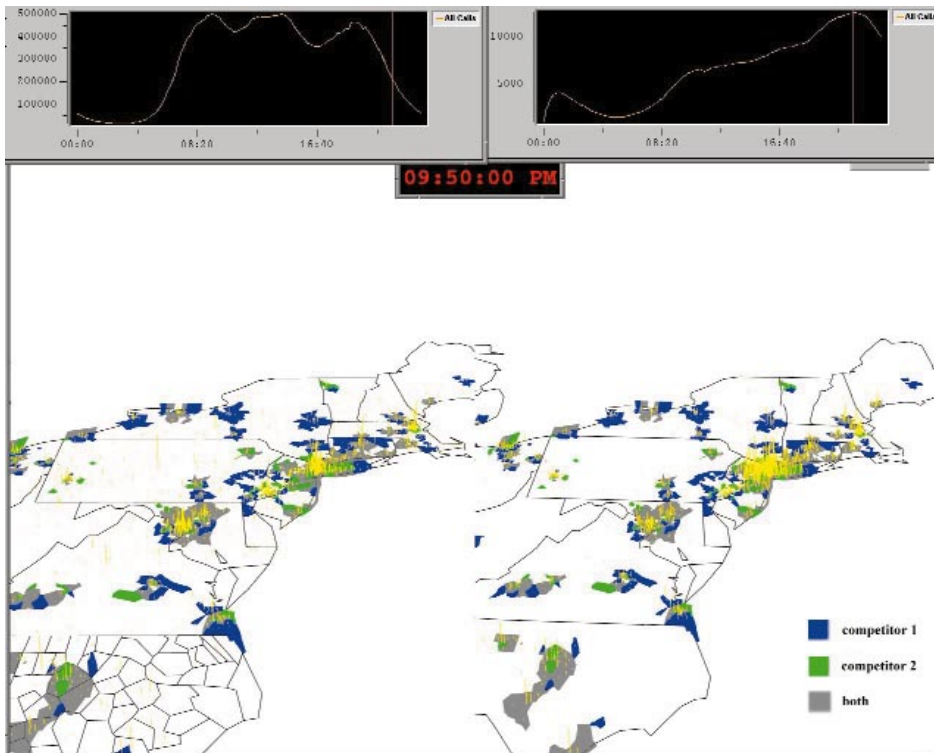
5 Market and service comparison between two competitors offering voice long-distance service and Internet access. The display shows information temporally and spatially. The left histogram and the yellow spike show volume for a voice service; the right histogram shows Internet access. Local calling areas are in green and blue (overlap is gray).

This type of visualization can help users understand the effect of geographic location on Internet access facilities and whether different services offered by the same company complement each other or compete.

Discussion and future work

As our examples suggest, using Swift-3D for network visualization, at scale, can qualitatively improve network management and marketing of network services. Although most of the individual screen shots aren't in themselves novel, they're interesting when running together in a time series on heavy telecommunications traffic. Of particular interest are Swift-3D's graphics and visualization for data access and drill-down queries.

Just as large-scale data management and networking have fundamentally changed how managers use information throughout an organization, high-performance graphics and visualization are changing how managers present and understand information in large telecommunication networks and services.

A key problem we are now addressing is how to show more information. Specifically, we're looking at how to combine multiple networks and services, such as voice, IP, asynchronous transfer mode (ATM) or frame relay, and wireless communication; and how to represent higher-order structures, such as virtual private networks and endpoints.

Classic work by Richard Becker, Stephen Eick, and Allan Wilks[2] shows how to attack scale in network visualization using techniques such as selection, aggregation, and thresholding. To this list of techniques, we add zooming and level-of-detail control. The need to visualize more abstract structures has grown. Techniques from computational geometry (such as graph drawing[3]), and mathematical and scientific visualization may be appropriate.

Acknowledgement

1. D.A. Keim and A. Hermann, "The Gridfit Algorithm: An Efficient and Effective Algorithm to Visualizing Large Amounts of Spatial Data," *Proc. IEEE Visualization Conf.*, IEEE Computer Society Press, Los Alamitos, Calif., 1998, pp. 181-188.
2. R.A. Becker, S.G. Eick, and A.R. Wilks, "Visualizing Network Data," *IEEE Trans. on Visualization and Computer Graphics*, Vol. 1, No. 1, March 1995, pp. 16-28.
3. G. Di Battista et al., *Graph Drawing: Algorithms for the Visualization of Graphs*, Prentice-Hall, Upper Saddle River, N.J., 1999.

Readers may contact Koutsofios and North at Information Visualization Research, AT&T Laboratories, 180 Park Avenue, Florham Park, N.J., 07932-0971; {ek, north}@research.att.com.

Contact department editors Rhyne and Treinish by e-mail at trhyne@vislab.epa.gov and lloydt@us.ibm.com.