# Visualizing Large-Scale Telecommunication Networks and Services

Eleftherios E. Koutsofios, Stephen C. North, Russell Truscott, Daniel A. Keim
Information Visualization Research, AT&T Laboratories, Florham Park, N.J., USA 07932-0971
e-mail: {ek, north, truscott, keim}@research.att.com

## Abstract

Visual exploration of massive data sets arising from telecommunication networks and services is a challenge. This paper describes SWIFT-3D, an integrated data visualization and exploration system created at AT&T Labs for large scale network analysis. SWIFT-3D integrates a collection of interactive tools that includes pixel-oriented 2D maps, interactive 3D maps, statistical displays, network topology diagrams and an interactive drill-down query interface. Example applications are described, demonstrating a successful application to analyze unexpected network events (high volumes of unanswered calls), and comparison of usage of an Internet service with voice network traffic and local access coverage.[1]

## 1. INTRODUCTION

Global telecommunication networks and services are among the enterprises having the highest volumes of real-time data. A voice network may complete more than 250 million calls per day. Each is described by one or more events, yielding a total of tens of gigabytes of data daily. Wireless, Asynchronous Transfer Mode (ATM), frame relay, Internet Protocol (IP) networks and higher-level services on them also are described by massive data sets, and can present additional problems in reconstructing an end-to-end view of user activity. Understanding this data at full scale is crucial for managing networks and improving their performance and reliability from a customer's viewpoint. Visualization techniques have become increasingly important to achieving this goal. In the *AT&T Infolab*[2], a new visualization system, *SWIFT-3D* has been created for interactive network data exploration. It incorporates interactive 3D maps, statistical displays, network topology diagrams, and pixel-oriented displays (see [7] for a brief overview).

Its applications include monitoring and analyzing activities at the network element, network-wide, customer and service levels. These activities may be network generated (e.g., exploration of network events and alarms or customer generated (e.g., usage anomalies such as fraud). End uses of the analysis would likely include improvement of service to customers, market analysis and gaining an understanding of previously hidden relationships between and within data segments.

## 2. TELECOMMUNICATION APPLICATIONS

The goal of this work is to support interactive visual exploration of databases that describe full-scale commercial telecommunication networks, and to simultaneously raise the level of abstraction in visualization, for example showing layered services or network performance from an individual customer's viewpoint. Derived from this goal is the ability to move from data to business decision within minutes.

Many data analysis tasks that are tractable on small or medium-sized data sets can be difficult at greater scale. When practitioners refer to terabyte databases, they sometimes mean databases of image, sound or video data. In contrast, our application involves working with many small records describing transactions and network status events. The data processing involved is different in terms of the number of records and data items to be interpreted. In voice networks, the detail record for each call conforms to an industry standard format (Automatic Message Accounting, or **AMA**) that has about 50 attributes such as originating and terminating phone numbers, date, time and duration of the call. In our application this information is stored for each of the hundreds of millions of calls made daily, yielding about 15 GByte of data uncompressed. In addition, data is collected from the other networks previously mentioned. Understanding the relationships between them is increasingly important, e.g. to manage integrated communication services for global enterprises, but the data management problems that result are even more challenging than for a single service.

More than just scale is involved: our goal is also to raise the level of abstraction in network visualization, and to improve the real-time response of our analyses. This can help network managers and business decision makers to recognize and respond to changing conditions quickly; within minutes when possible. A scalable research prototype for visually exploring full-scale network traffic must therefore provide good interactive response, avoid instance-specific processing, and be flexible enough to support experiments in both back-end queries and the user interface. In our initial experiments, we found that commercial database systems either couldn't handle such large volumes or consumed far too many resources. Another problem with commercial databases is that the administrative effort was too high to support experimental research. Databases, however, have many useful features, such as data independence, and a standard query language. The tools we have built have some of these features. A main difference in our approach is the emphasis on data streaming, in comparison to a query/response methodology employed by formal databases.

[1] The examples in this paper are only for illustration and not intended to represent any specific service, customer or competitor of AT&T.

[2] The *InfoLab* was formed in *AT&T Research* in 1996 as an interdisciplinary project to support research in analysis of massive network-related data. Current projects rely on the resources of several SGI Origin-2000 servers, 5 terabytes of disk, and an SGI Onyx connected to Powerwalls for visualization research.

# 3. VISUALIZATION IN *SWIFT-3D*

The *SWIFT-3D* system integrates a collection of relevant visualization techniques ranging from familiar statistical displays, to pixel-oriented overviews with interactive 3D-maps and drag+drop query tools. It provides comprehensive support for data exploration, integrating large scale data visualization with querying, browsing and statistical evaluation (see [1,2,3,8] for examples of previous related work). The visualization component maps the data to a set of linked 2D and 3D views [4,9] created by different visualization techniques:

- ***Statistical 2D Visualizations*** (line graphs, histograms, etc.) - used as overview displays and for interactive data selection
- ***Pixel-oriented 2D Visualizations*** - intended as bird's-eye overviews and for navigation in 3D displays
- ***Dynamic 3D Visualizations*** - used for an interactive detailed viewing of the data from different perspectives.

In addition, the system provides tightly integrated *browsing and querying tools* to select the data to be displayed and to drill-down for details if some interesting pattern has been found.

A screenshot of the overall system is given in Figure 1. The upper left window shows a time line visualization of voice network volume in 10-minute intervals. This plot shows the volume for different services (e.g. residential, business, and 1+ dial-around service, software-defined networks, and aggregate volume). The window below the time line allows the user to select data for display by date, time or type of service. The large window shows a three-dimensional display of the data using a histogram spike for each location to display a value (typically, level of activity) corresponding to the cursor position in the time line window (11:00). The user can interactively navigate in the 3D display, zoom in at interesting locations, or view the map from arbitrary perspectives. An automated path-planning module has also been designed to determine a natural, context-preserving path from one viewpoint to another. The mapping between the data and display objects is set in an auxiliary file that contains geometric information about points, lines, polygons, and triangles, and coloring. Various color maps may be defined to highlight interesting properties of data. The mapping file may contain multiple levels of detail; for example, a data set representing the United States may be divided according to state, county, and telephone exchange, census block and 9-digit postal zip code outlines. Also, multiple data value sets can be mapped to the same geometry. For example, we can map state population to the state outline level and county population to the county level. As the view of a state enlarges, the displays can shift from showing a single value for state population to showing one per county. The user may also play through an adjustable interval in the time line window to get an animated time-sequence display (see video). If the user sees an interesting pattern in the visualization window, a drag-and-drop interface is available to drill-down to get details, explore context and take actions if necessary. This provides an intuitive way of converting spatial information into detailed information such as the top originating or top dialed numbers.

An additional 2D overview window is provided, showing call volume for each location by one colored pixel (cf. lower left corner of Figure 1). The technique behind the pixel-oriented 2D overviews is an adaptation of the *Gridfit* approach used in the VisualPoints system [6]. *Gridfit* places data points on a pixelated display, so that points having coordinates that would normally map to the same display pixel are represented by other pixels that would otherwise be unoccupied. Its algorithm is based on hierarchical partitioning of the data space, using a top-down reallocation of the screen space according to the requirements of subregions. Gridfit allows an efficient and effective repositioning of the pixels on the screen such that the (absolute and relative) position of the data points and their distance is preserved as much as possible. The color is chosen such that high call volumes are mapped to dark colors and low call volumes are mapped to bright colors.

# 4. IMPLEMENTATION ARCHITECTURE

*SWIFT-3D* consists of three modules: data collector, aggregator, and visualization interface. These communicate using self-describing data-independent binary formats consisting of a header that defines record size, type, and data context, followed by the actual data. This is necessary since *SWIFT-3D* is designed to work in real-time: the data processing modules can work incrementally and the visualization tools can safely access the data files while they are being updated.

To achieve good performance, *SWIFT-3D* uses various techniques that minimize processing delays and the use of system resources. Such techniques include processing pipelines, direct IO, memory mapping, and dynamic linking of on-the-fly generated code.

## 4.1 Data Collection and Storage

*SWIFT-3D* must collect data from many different sources having their own specialized formats. *SWIFT-3D* includes tools to convert such data to its internal self-describing format. When data is already in a fixed format, all that needed is to associate a data record schema with the file. In most cases, standard tools suffice for data conversion, but some types of data need more intricate pre-processing. For example, the voice network generates records in the previously-mentioned AMA format. This format has many sub-record types that can be combined to describe a call. Extracting information from AMA files is even further complicated because depending on type of call, a value can be stored in different sub-records. For example, the dialed number is kept in different places in domestic and international calls. Such idiosyncratic processing is performed by custom tools to load into *SWIFT-3D* format.

## 4.2 Data Processing

Initial processing of a data feed usually involves reading in records and computing basic statistics. *SWIFT-3D* relies on a stream pipeline model. Accessing large-scale data on disk can be expensive, so instead of storing the output of each processing step to disk, the stream processors are implemented as concurrent processes that exchange data. *SWIFT-3D* extends the UNIX pipe model of single writer and single reader to that of single writer and many readers to minimize the amount of data copying. For example, we might need to process a day's worth of telephony data and compute: (1) how many calls there were per area code and exchange (NPA/NXX), (2) how many calls did not complete and separate these by failure type. This could be implemented by a single process that reads data from disk and feeds two other processes to count.

*SWIFT-3D* provides several tools to be used in such pipelines. These include tools to filter records (e.g. remove calls that did not complete), count based on attributes (e.g. count number of incomplete calls by NPA/NXX), split a single file into several based again on some combination of attributes (e.g. separate calls into a file per type of service such as toll free calls, operator calls, collect calls, etc.).

The expressions used for filtering, counting, and splitting are specified as C-style expressions. For example, the expression ’{ if (tos == TOLLFREE && iscomplete) KEEP; else DROP; }’ could be used to filter out calls that are not toll free (1-800, 1-888, 1-877) or not complete. These expressions are turned into C code that is compiled into shared objects on the fly and are then dynamically linked in and executed. This approach combines the speed of compiled code with the flexibility of tools such as AWK. C and AWK seem to be the tools of choice for most of our statisticians and analysts.

## 4.3 Data Visualization

*SWIFT-3D*’s visualization tools allow users to explore data filtered by the stream processors. They are designed to be interactive in the sense that the user can view some data set, focus on a specific subset, query the system for the raw data that generated this subset, re-aggregate and view the new result. *SWIFT-3D* enables this by keeping enough information to link raw data, aggregate data and visual objects. This link to visual objects is implemented by generating geometric data sets that contain information about the items they represent. For example, an NPA/NXX may be represented by a point, bar, or polygon of its geographic area. In all cases, the geometry file contains information to link the NPA/NXX to the point, line, or polygon. Besides being used to answer a user query, this facility is also used to alter the geometry based on data values. For example, if NPA/NXXs are shown as polygons and busy NPA/NXXs need to be colored red, the system uses this mapping to determine red polygons.

For reading records off disks, *SWIFT-3D* uses Direct-IO if available. Direct-IO bypasses kernel buffer copying from disk, and can be twice as fast as normal IO. (Normal IO can be faster for data that was recently read and is still in cache, but given the size of our datasets, this is rarely the case).

The format of the counts files is also self-describing. Such files implement a type of two dimensional array of values (integers, floats, etc.). One dimension (the `frame’) corresponds to time buckets while the other (the `item’) corresponds to the aggregation type. The second dimension can be accessed using a dictionary that maps item ids to item positions. For example, the second dimension in the scenario above may have an entry per NPA/NXX observed, and the dictionary might indicate that data for NPA/NXX 973-360 is in position #0.

These files are designed to be accessed and changed incrementally: when new data arrives, these files are opened and the various counts are increased in place (using some amount of buffering to minimize accesses to the files). The actual updating of the files is done using memory mapping, due to the random access nature of the updating. File locking is used to protect against accessing such a file in the middle of an update. Also, each update increments a count stored in the file. This allows the visualization tools to efficiently check if the file has been modified.

## 5. APPLICATIONS

*SWIFT-3D* has been applied to several different problems in network visualization. These include the ability to provide an abstraction that permits visualization of the data across the information strata of network element, network, services and customers; the ability to view cross network interactions and their impact upon a service and/or customer; the capabilities to discern impact on one or more customers when there is a network event.

An interesting example is the examination of calls that cannot be completed due to congestion at the customer premise. Keeping this number low is important due to the resources consumed. This is important both to the customers (who need reliable service for telemarketing sales and customer support) and to a network/service provider from a financial standpoint (unanswered calls consume network resources and incur cross-carrier settlement charges without creating revenue). In visually exploring voice network events, we noticed that on several days within an interval of several weeks, many unanswered calls originated in a certain metropolitan area (cf. Figure 2). The events always occurred at bottom of the hour (:30) for several hours in the evening. By interactive querying we found that most of the calls were directed at one 800 number, and that the number belonged to a radio station. By tuning in, we discovered that the station was giving out free tickets for an upcoming concert. The winner was the tenth caller at the bottom of each hour.

Another application concerns analysis of an Internet service. There is considerable motivation for understanding relationships between usage of multiple services, both from a single service provider, and between competitors. *AT&T* wanted to know how much coverage an Internet access service had. The coverage is measured by the number of the area code and exchanges (and ultimately households or customers) where connecting to the Internet is a local call (usually without per-minute charges). We contracted two companies that claimed to have such data. We gave them the locations of the modem pools and asked them to tell us what codes and exchanges were covered. We received two very different answers. To understand the differences we used *SWIFT-3D*: areas claimed to be covered in the answer of company A were colored blue, those claimed to be covered in the answer of Company B were colored green, where both companies agreed, the map was colored gray. We noticed widespread differences in many states, while a few states had good matches.[1] In order to decide which company’s answer was more correct, we superimposed our customer usage data on the map. In the generated visualizations (cf. Figure 3), we saw that there was a lot of usage in gray and blue areas, but very little usage in green (and almost none in black areas). Our conclusion was that the answer by company A was more correct. It further became clear that individual customers are very aware of local calling areas, and are not willing to use an ISP when the access would be too expensive. A side effect of our findings was that the business decided to not even advertise this service in areas not covered by local access.

A third application involves recognizing the characteristics of virtual private networks (VPNs) provisioned by customers on a large packet network, and their relationships to physical network facilities. Figure 4 shows the peak volume of Permanent Virtual Circuit (PVC) traffic, by VPN, for the whole network in one 5-minute period. The display highlights the PVCs having the greatest load. The eventual goal of this study is to understand customer-focused, near real-time management of packet networks.

[1] The confusion is actually caused because individual phone companies have differing schemes for providing local calls.

## 6. FUTURE WORK

Although this project has yielded considerable capabilities there are facets of the current work that we will be looking at in the coming year.

Visualization for the masses: Our current environment (an 8-projector display wall driven by a Silicon Graphics Onyx server) provides “visualization for the mass” - that is, the mass of people standing around viewing it. It would be advantageous to move network visualization to an environment that provides “visualization to the masses.” To do this, we are prototyping some of *SWIFT-3D*’s key modules in web environments such as Java3D.

Extending beyond a network view: To date, network visualization as practiced in industry has been rooted in a network element perspective. Focusing on an end-to-end view of an entire customer, reaching all the way to individual endpoint devices requires re-examining the visualization metaphors and abstractions. Achieving this goal also depends upon acquiring and processing much more data than we currently do. A reasonable estimate is that approximately 50GB of data will be collected per day.

Extending the information strata: The structure of the information strata was based upon the Telecommunications Management Network (TMN) standards. In TMN, the Logical Layered Architecture (LLA) describes a stratification of network element, network, service and business. We have extended this model to include customers but believe we will need yet another layer that refers to an industry. We expect that industries will be shown to have differing use patterns and that the visualization of these patterns will lead us to new understandings.

Novel network display metaphors: New applications, such as visualization of services on wide-area IP networks involve looking at flows between endpoints and other higher-order structures. An open problem is how to visualize higher-order objects in large networks. Approaches from graph drawing are relevant [5], but there is currently no visualization technique that is completely satisfactory for viewing the structure of large, general graphs. The best proposed techniques amount to imposing some hierarchical structure, or filtering graphs to make them much smaller. We plan to adapt some of these techniques to our applications.

## 7. CONCLUSIONS

Effective visual exploration of massive telecommunication data sets requires tightly integrating a diverse collection of visualization and analysis tools and techniques. Each of the applications we tried has different requirements, and so it is valuable to have a flexible environment for experiments on scalable prototypes. In using the system, users often observe interesting aspects in an overview visualization and then explore them by means of detailed visualizations, drill-down facilities, and drag-and-drop queries. Except in the most simple situations, visualization is not a closed, linear process; exploration seems to be inherently non-linear and therefore the ability to switch easily between techniques is very important. In addition, interactive processing of network data at full scale is crucial to many applications.

## Acknowledgements

The authors thank David Belanger, Ken Church and Manolis Tsangaris for their collaboration on network visualization, and especially with help to obtain experimental data.

## References

[1] Ahlberg, C. and E. Wistrand. ‘IVEE: An Environment for Automatic Creation of Dynamic Queries Applications’, Proc. ACM CHI Conf. on Human Factors in Computing (CHI95), Demo Program, 1995.

[2] Ahlberg, C. and E. Wistrand. ‘IVEE: An Information Visualization and Exploration Environment’, Proc. Int. Symp. on Information Visualization, Atlanta, GA, 1995, pp. 66-73.

[3] Becker, Richard A., Stephen G. Eick and Allan R. Wilks. *Visualizing network data*. IEEE Transactions on Visualization and Computer Graphics, 1(1), pp. 16-28, March 1995.

[4] Buja A., J.A. McDonald, J. Michalak and W. Stuetzle. *‘Interactive Data Visualization Using Focusing and Linking’*, Visualization ‘91, San Diego, CA, 1991, pp. 156-163.

[5] Di Battista, Giuseppe, Peter Eades, Roberto Tamassia and Ioannis Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*, Prentice-Hall, 1999.

[6] Keim, Daniel A. and Annemarie Herrmann. *The Gridfit Algorithm: An Efficient and Effective Algorithm to Visualizing Large Amounts of Spatial Data*, IEEE Visualization Conference, Research Triangle Park, NC, pp. 181-188, 1998.

[7] Koutsofios, Eleftherios E., Stephen C. North and Daniel A. Keim. *Visual Exploration of Large Telecommunication Data Set*s, Visualization Blackboard, IEEE Computer Graphics and Applications, May 1999, to appear.

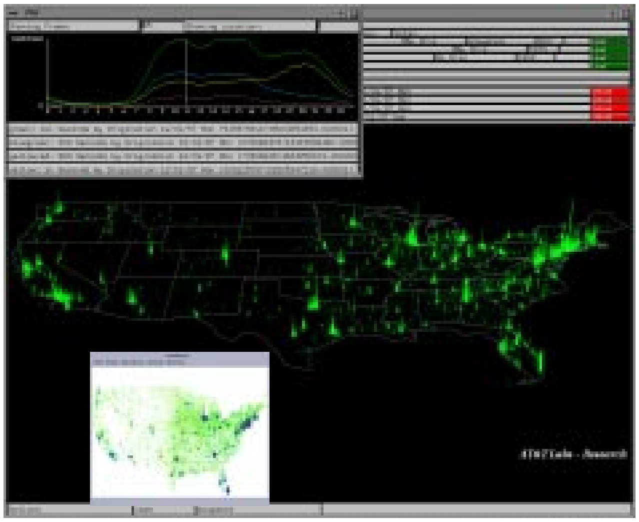
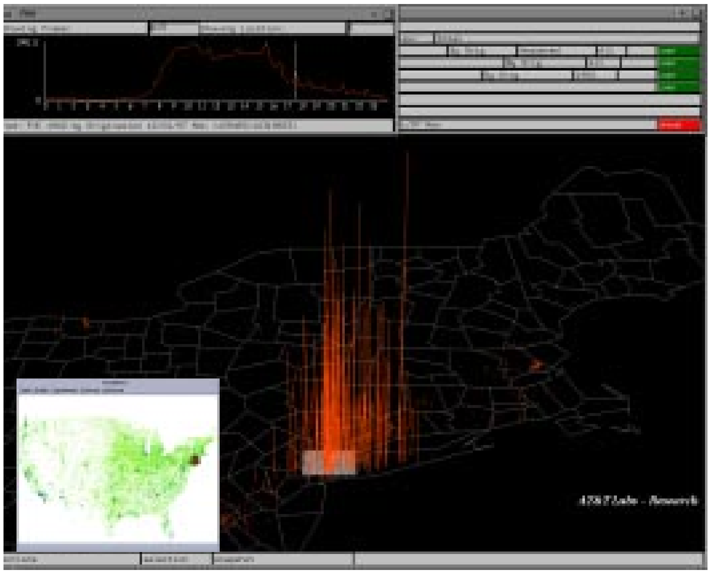


**Figure 1. Swift-3D overview.**

**Figure 2. Inspection of network event's effect on customers.**



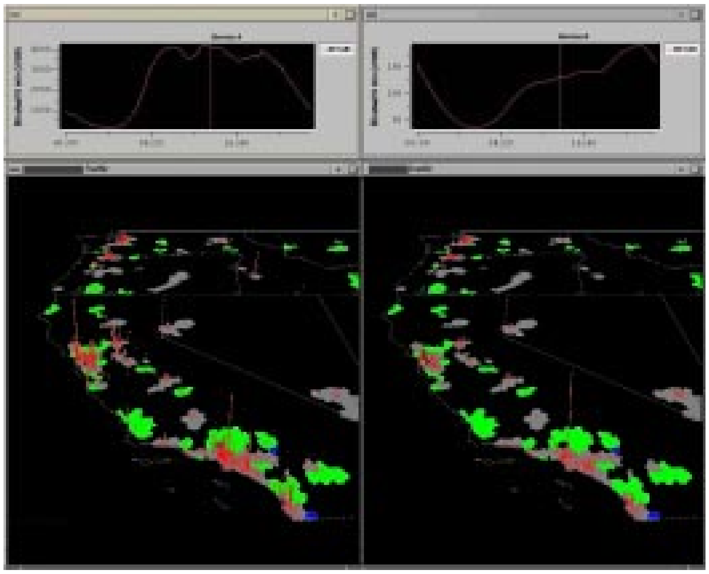
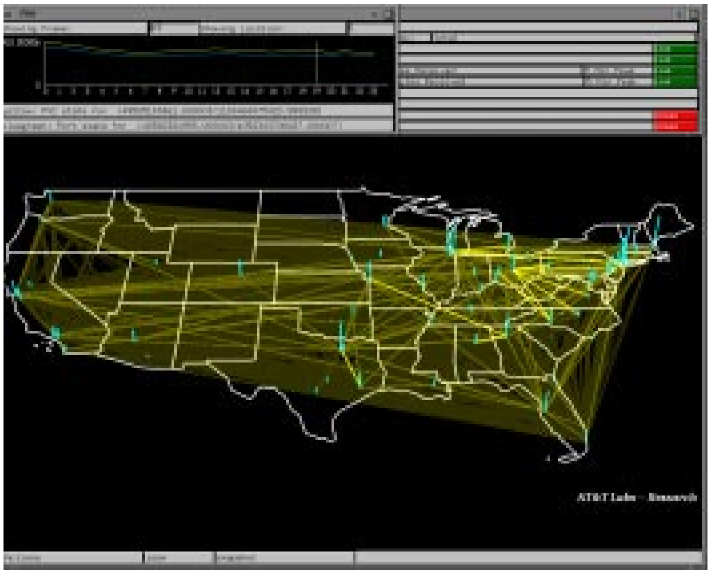


**Figure 3. Market and service comparison.**

**Figure 4. Virtual private network activity.**


AT&T Labs - Research