First publ. in: User interfaces to data intensive systems proceedings (UIDIS'99), 5 - 6 September 1999, Edinburgh, Scotland / Ed. by Norman W. Paton ... Los Alamitos: IEEE Computer Soc., 1999, pp. 12-20

Visual Exploration of Large Telecommunication Data Sets

Daniel A. Keim
Database and Visualization Group, University of Halle, Kurt-Mothes-Str. 1, 06120 Halle, Germany
e-mail: keim@informatik.uni-halle.de

Eleftherios E. Koutsofios, Stephen C. North
Information Visualization Research, AT&T Laboratories, Florham Park, N.J., USA 07932-0971[1]
e-mail: {ek, north}@research.att.com

Abstract

Visual exploration of massive data sets arising in the telecommunication industry is a challenge. This paper describes a number of different techniques for visually exploring large data sets. The techniques cover a wide range of techniques, including statistical 2D displays, pixel-oriented displays, and dynamic 3D displays with variable resolution. The techniques have been successfully applied in the telecommunications industry to analyze call detail data for understanding customer behavior and preventing fraudulent usage, and to monitor network traffic for analyzing unexpected network events such as high volumes of unanswered calls.

1. Introduction

The progress made in hardware technology allows today's computer systems to store very large amounts of data. The available storage space is easily filled with data which is often automatically recorded via sensors and monitoring systems. Today, even simple transactions of every day life, such as paying by credit card or using the telephone, are typically recorded by using computers. Usually, many parameters are recorded, resulting in multidimensional data with a high dimensionality. The data of all areas mentioned so far is collected because people believe that it is a potential source of valuable information providing a competitive advantage (at some point). Finding the valuable information hidden in them, however, is a difficult task. With today's database systems and their query tools, it is only possible to view quite small portions of the data. If the data is presented textually, the amount of data which can be displayed is in the range of some one hundred data items, but this is like a drop in the ocean when dealing with data sets containing millions of data items. Having no possibility to adequately query and view the large amounts of data which have been collected because of their potential usefulness, the data becomes useless and the database becomes a data 'dump'.

Global telecommunication network and service companies 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 are also 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], new visualization techniques are used for interactive network data exploration. The techniques used include interactive 3D maps, statistical displays, network topology diagrams, and pixel-oriented displays. Its applications include monitoring and analyzing activities at the network element, network-wide, customer and service level. 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.

The goal of the work described in this paper 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, the telecommunication 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.

In this paper, we provide an overview of the techniques used for visually exploring large telecommunication data sets. Section 2 briefly introduces some techniques for visualizing large multidimensional data sets and section 3 deals with techniques that specifically include time- and geometry-related information. In section 4, we describe a number of example applications of the techniques at AT&T including fraud detection, network monitoring, and call volume analysis. Section 5 provides some concluding remarks.

2. Visualization of Multidimensional Data

In telecommunication applications, there are a number of multidimensional data sets. For these data sets, there is no standard mapping into the Cartesian coordinate system since the data does not have some inherent two- or three-dimensional semantics. In this section, we provide an overview of multidimensional visualization techniques and provide a classification of the existing techniques (cf. subsection 2.1). Then, we discuss one category - the pixel-oriented techniques - in a little more detail (cf. subsection 2.2).

2.1 Classification of Multidim. Visualization Techniques

There are a number of well-known techniques for visualizing multidimensional data sets which can be best classified using three orthogonal criteria: the visualization technique, the distortion technique, and the interaction technique (cf. Figure 1). Orthogonality means in this context that any of the visualization techniques can be used in conjunction with any of the distortion as well as any of the interaction techniques. The visualization techniques can be divided into geometric projection, icon-based, pixel-based, hierarchical, and graph-based techniques. Well-known examples of geometric projection techniques include scatterplot matrices and coplots [And 72, Cle 93], landscapes [Wri 95], prosection views [FB 94, STDS 95], hyperslice [WL 93], and parallel coordinates [Ins 85, ID 90]; examples of icon-based techniques are stick figures [PG 88], shape-coding [Bed 90], and color icons [Lev 91, KK 94]; examples of pixel-oriented techniques are the spiral [KK 94, Kei 96], recursive pattern [KKA 95] and circle segment techniques [AKK 96]; examples of hierarchical techniques are dimensional stacking [LWW 90], treemap [Shn 92, Joh 93], and cone-trees [RMC 91]; and examples of graph-based techniques are cluster- and symmetry-optimized

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 storage, and an SGI Onyx connected to Powerwalls for visualization research.

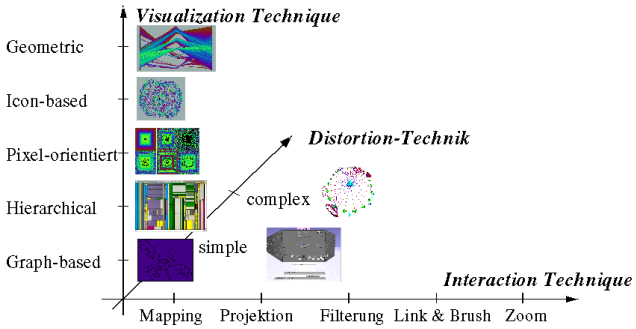

Figure 1: Classification of Multidimensional Visualization Techniques

as well as hierarchical graph visualizations [BEW 95, BETT 99]. In addition to the visualization technique, for an effective data exploration it is important to use some interaction and distortion techniques. The interaction techniques allow the user to directly interact with the visualization. Examples of interaction techniques include interactive mapping [BF 93, BSC 96], projection [Asi 85, BCS 96], filtering [AS 94, Eic 94, FS 95], zooming [Bed 94, AW 95], and interactive linking and brushing [War 94, WUT 95]. Interaction techniques allow dynamic changes of the visualizations according to the exploration objectives, but they also make it possible to relate and combine multiple independent visualizations. Note that connecting multiple visualizations by linking and brushing, for example, provides more information than considering the component visualizations independently. The last criterion of the classification helps in the interactive process of exploration by providing means for focussing while preserving an overview of the data. The basic idea of distortion techniques is to show portions of the data with a high level of detail while others are shown with a much lower level of detail. A number of simple and complex distortion techniques may be used for this purpose [LA 94]. Examples are the perspective wall [MRC 91], bifocal lens [AS 82], table lens [RC 94], fisheye view [Fur 86, SB 94], hyperbolic tree [LR 94, LRP 95, MB 95], and hyperbox techniques [AC 91].

This brief introduction of our classification and the enumeration of examples is aimed at providing a more structured understanding of the large number of available multidimensional visualization techniques. It can also be used as a starting point to compare the available techniques, to improve existing techniques, and to develop new techniques. To provide a starting point for such a comparison, in Figure 2 we provide a preliminary and subjective comparison table[3] which is trying to compare a number of visualization techniques. The comparison of the visualization techniques is based on their suitability for certain

- **data characteristics**
  such as number of dimensions (attributes), number of data items, and suitability for categorical data,

		clustering	multi-variate hot spot	no. of variates	no. of data items	categorical data	visual overlap	learning curve
Geometric Techniques	Scatterplot Matrices	++	++	+	+	-	o	++
	Landscapes	+	+	-	o	o	+	+
	Prosection Views	++	++	+	+	-	o	+
	Hyperslice	+	+	+	+	-	o	o
	Parallel Coordinates	o	++	++	-	o	--	o
Icon-based Techniques	Stick Figure	o	o	+	-	-	-	o
	Shape Coding	o	-	++	+	-	+	-
	Color Icon	o	-	++	+	-	+	-
Pixel-oriented Techniques	Query-Independent	+	+	++	++	-	++	+
	Query-Dependent	+	+	++	++	-	++	-
Hierarchical Techniques	Dimensional Stacking	+	+	o	o	++	o	o
	Treemap	+	o	+	o	++	+	o
	Cone Trees	+	+	o	+	o	+	+
Graph-based Techniques	Basic Graphs	o	o	-	+	o	o	+
	Specific Graphs	++	+	-	+	o	+	+

Figure 2: An Attempt of Comparing Multidimensional Visualization Techniques

- **task characteristics**
  such as clustering and multi-variate hot spots,
- **visualization characteristics**
  such as visual overlap and learning curve.

A more detailed description of the classification and examples can be found in tutorial notes on visual data exploration [Kei 97a, Kei 97b]. In the following, we introduce one class of our classification - the pixel-oriented techniques - in more detail.

2.2 Pixel-oriented Techniques

The basic idea of pixel-oriented techniques [KK 94] is to map each data value to a colored pixel and present the data values belonging to one attribute in separate windows (cf. Figure 3). Since in general our techniques use only one pixel per data value, the techniques allow us to visualize the largest amount of data, which is possible on current displays (up to about 1,000,000 data values). If each data value is represented by one pixel, the main question is how to arrange the pixels on the screen. Our pixel-oriented techniques use different arrangements for different purposes. If a user wants to visualize a large data set, the user may use a query-independent visualization technique which sorts the data according to some attribute(s) and uses a screen-filling pattern to arrange the data values on the display. The query-independent visualization techniques are especially useful for data with a natural ordering according to one attribute (e.g., time series data). However, if there is no natural ordering of the data and the main goal is an interactive exploration of the database, the user will be more interested in feedback to some query. In this case, the user may turn to the query-dependent visualization techniques which visualize the relevance of the data items with respect to a query. Instead of directly mapping the data values to color, the query-dependent visualization techniques calculate the distances between data and query values, combine the distances for each data item into an overall distance, and visualize the distances for the attributes and the overall distance sorted according to the overall distance. The arrangement of the data items centers the most relevant data items in the middle of the window, and less relevant data items are arranged in a spiral-shape to the outside of the window.

3. **Disclaimer:** The comparison table expresses my (Daniel Keim) personal opinion obtained from reading the literature and experimenting with several of the described techniques. Many of the ratings are arguable and largely depend on the considered data, the exploration task, experience of the user, etc. In addition, implementations of the techniques in real systems usually avoid the drawbacks of the single techniques by combining them with other techniques, which is also not reflected in the ratings.

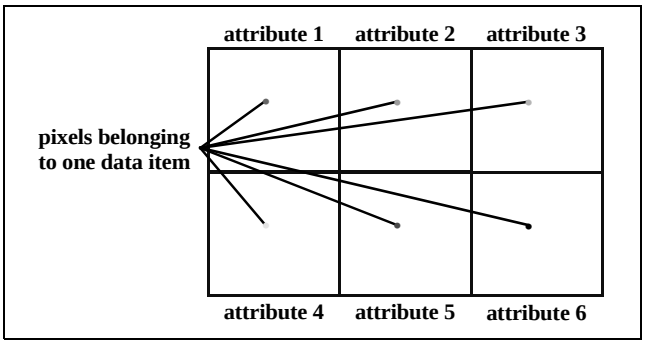

Figure 3: Arrangement of Attribute Subwindows for Data with Six Attributes

All pixel-oriented techniques partition the screen into multiple windows. For data sets with m attributes (dimensions), the screen is partitioned into m windows — one for each of the attributes. In case of the query-dependent techniques, an additional (m+1)th window is provided for the overall distance. Inside the windows, the data values are arranged according to the given overall sorting which may be data-driven for the query-independent techniques or query-driven for the query-dependent techniques. Correlations, functional dependencies, and other interesting relationships between attributes may be detected by relating corresponding regions in the multiple windows.

Query-Independent Pixel-oriented Techniques

Simple query-independent arrangements are to arrange the data from left to right in a line-by-line fashion or top-down in a column-by-column fashion. If these arrangements are done pixelwise, in general, the resulting visualizations do not provide useful results. More useful are techniques which provide a better clustering of closely related data items such as space-filling curves (e.g., the well-known curves by Peano & Hilbert [Pea 90, Hil 91] and Morton [Mor 66]). For data mining more important are techniques that provide nice clustering properties as well as an arrangement which is semantically meaningful. An example for a technique which has these properties is the recursive pattern technique [KKA 95]. The recursive pattern is based on a generic recursive scheme which allows the user to influence the arrangement of data items.

It is based on a simple back and forth arrangement: First, a certain number of elements is arranged from left to right, then below backwards from right to left, then again forward from left to right, and so on. The same basic arrangement is done on all recursion levels with the only difference that the basic elements which are arranged on level i are the patterns resulting from level(i-1)-arrangements. Let w_i be the number of elements arranged in the left-right direction on recursion level i and h_i be the number of rows on recursion level i. On recursion level i ($i \geq 1$), the algorithm draws w_i level(i-1)-patterns h_i times alternately to the right and to the left. The pattern on recursion level i consists of $w_i \times h_i$ level(i-1)-patterns, and the maximum number of pixels that can be presented on recursion level k is given by $\prod_{i=1}^{k} w_i \times h_i$. Examples of recursive pattern visualizations are provided in section 4.

Query-Dependent Pixel-oriented Techniques

The idea of the query-dependent visualization techniques [KK 94, Kei 96] is to visualize the data in the context of a specific user query to give the users feedback on their queries and direct their search. Instead of directly mapping attribute values to colors, the distances of attribute values to the query are mapped to colors. To describe the idea of the query-dependent techniques, we view the relations of a relational database as sets of tuples $(a_1, a_2, ..., a_k)$ with $a_1, a_2, ..., a_k$ denoting the attribute values of a data item. Simple queries against the database can be described as regions in the k-dimensional space defined by the k attributes of the relation. If exactly one query value is specified for each attribute, the query corresponds to a point in k-dimensional space; if a query range is specified for each attribute, the query corresponds to a region in k-dimensional space. The data items which are within the query region form the result of the query. In most cases, the number of results cannot be determined a priori; the resulting data set may be quite large, or it may even be empty. In both cases, it is difficult for the user to understand the result and modify the query accordingly. To give the user more feedback on the query, our visual data mining techniques do not only present the data items which are within the query region, but also those which are 'close' to the query region and only approximately fulfill the query. For determining the approximate results, distances between the data and query values are calculated. The distance functions are data type and application dependent. For numeric types such as *integer* or *real* and other metric types such as *date,* the distance of two values is easily determined by their numerical difference. For other types such as *strings,* multiple distance functions such as the lexicographical difference, character-wise difference, substring difference, or even some kind of phonetic difference may be useful. The distance calculation yields distance tuples $(d_1, d_2, ..., d_k)$ which denote the distances of the data to the query. We extend the distance tuples by a distance value d_{k+1}, denoting the overall distance of a data item to the query. The value of d_{k+1} is zero if the data item is within the query region; otherwise d_{k+1} provides the distance of the data item to the query region. In combining the distance values $(d_1, d_2, ..., d_k)$ into the overall distance value d_{k+1}, user-provided weighting factors $(w_1, w_2, ..., w_k)$ are used to weight the distance values according to their importance. The distance tuples $(d_1, d_2, ..., d_k, d_{k+1})$ are sorted according to the overall distance d_{k+1}. Then the distance tuples are mapped to color. In this step, the value ranges for each of the attributes and for the overall distance are mapped to a colorscale which has been specifically designed for our visual data mining techniques. Note that the human visual system has a non linear response to luminance and spectral content. Incorrect use of color can hide existing relations between variables, and introduce artifacts. It is therefore important to use a colorscale which is perceptually equally spaced [HL 92]. Our colorscale uses yellow to depict the distance 'zero' and a decreasing lightness to depict increasing distance values. The colors for approximate results range from green over blue and red to almost black. For details about our color mapping, the reader is referred to [KK 95b].

Since the focus of the query-dependent techniques is on the relevance of the data items with respect to the query, different arrangements of the pixels are appropriate. In experimenting with different arrangements, we found that for visualizing the results of a database query it seems to be most natural to present the data items with highest relevance to the query in the center of the display. With decreasing relevance with respect to the query the data pixels have to be placed further away from the center. The overall distance is used to place the data items on a spiral moving from the center to the outside. To retain the local clustering of data pixels which gets lost on a one-pixel-wide spiral, the spiral shape arrange-

ment is combined with a local 2,4, or 8 pixel-wide Peano-Hilbert curve.

As for the query-independent visualization techniques, a separate visualization for each of the selection predicates (attributes) and the overall distance is generated. In all subwindows, we place the pixels for each data item at the same position as the overall distance for the data item in the overall distance subwindow. By relating corresponding regions in the different windows, the user is able to perceive data characteristics such as multidimensional clusters or correlations. Additionally, the separate windows for each of the selection predicates provide important feedback to the user; for example, on the restrictiveness of each of the selection predicates and on single exceptional data items. Examples of spiral visualizations are provided in section 4.

All mentioned pixel-oriented techniques are implemented as part of the VisDB system. The details of the VisDB system as well as other pixel-oriented techniques can be found in [KKS 94, KK 95a, KK 95b, KK96].

3. Visualization of Time- and Geometry-related Data

In addition to visualization techniques for arbitrary multidimensional data, telecommunication applications also deal with large geometry-related data sets. For geometry-related information (in telecommunication data sets are usually 2D) there are a number of well-established visualization techniques such as maps, graphs, etc. The problem is that in general these techniques do not scale to the extend required for large scale telecommunication data sets. Therefore, the techniques have to be extended to allow an effective support of large scale data exploration.

The *SWIFT-3D* system, developed at AT&T research labs [KNK 99, KNTK 99], integrates a collection of relevant visualization techniques ranging from familiar statistical display 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 [AW 95, BEMW 95, RLG 96] for examples of previous related work). The visualization component maps the data to a set of linked 2D and 3D views [BMMS 91, War 94] 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.

The ***statistical displays*** (e.g., line graphs and histograms) are used to provide an overview of the data. Usually, a line graph or histogram showing the development of some parameter (e.g., call volume) aggregated over time is used as an overview display (cf. upper left corner of Figure 4). The statistical graphs are used to interactively select a certain time step to be displayed in the other visualizations. The ***pixel-oriented 2D displays*** provide a more detailed overview of the data, showing the call volume for each location by one colored pixel (cf. lower left corner of Figure 4). The technique behind the pixel-oriented 2D displays is an adaptation of the *Gridfit* approach used in the VisualPoints system [KH 98]. The *Gridfit* algorithm 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. The 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. The ***dynamic 3D displays*** provide a detailed view of the data. They show a histogram spike for each location to display a value corresponding to the cursor position in the time line window. 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

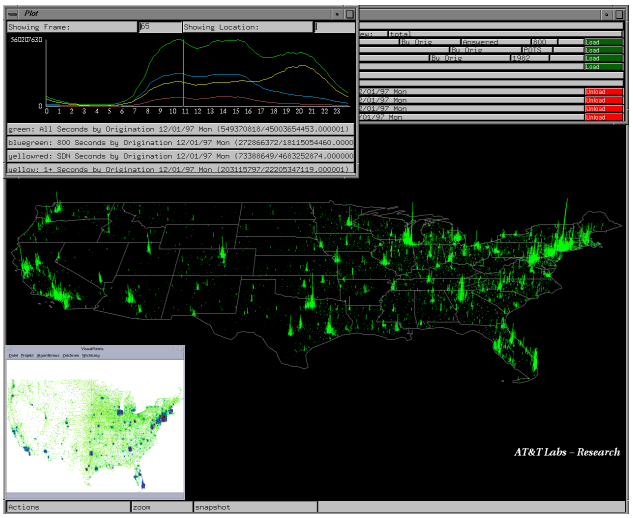

Figure 4: Swift-3D overview.

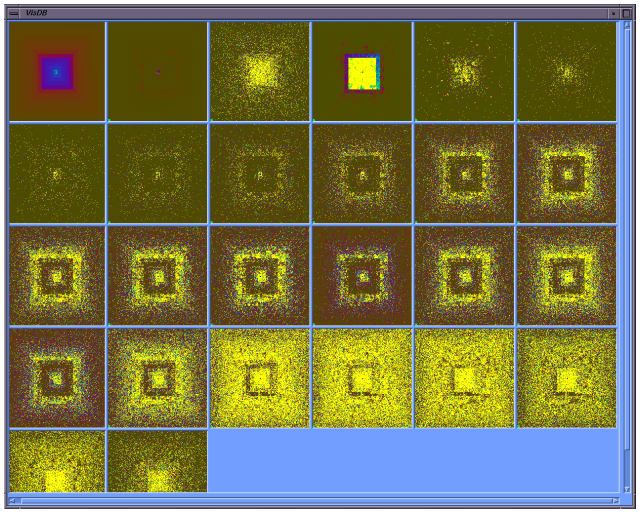

Figure 5: Query-dependent Visualization of Call Detail Data

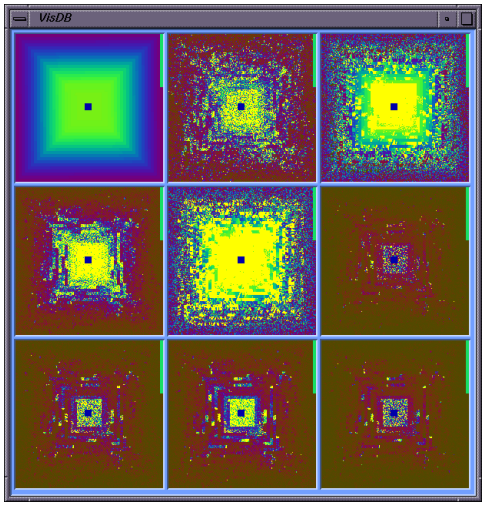

Figure 6: Query-dependent Visualization of WorldNet Data

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.

A screenshot of the overall system is given in Figure 4. 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 pixel-oriented overview display is shown in the lower left corner of Figure 4. The large window shows a three-dimensional display of the data using a histogram spike for each location to display the call volume corresponding to the cursor position in the time line window (11:00). The user can interactively navigate in the 3D display and view the map from arbitrary perspectives. The user may also play through an adjustable interval in the time line window to get an animated time-sequence display. 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.

4. Applications in Telecommunication

The techniques introduced in the previous subsections have been successfully applied on a number of different telecommunication data sets[4]. The *query-dependent spiral-shaped visualizations* are, for example, used for fraud detection in call detail data. The data set used for this purpose contains the calling time of a number of customers aggregated for each hour of the day. This means that the data sets consists of 24 time bins (plus some additional information) containing the calling time within the corresponding time period. In the query-dependent visualizations it is now interesting to focus on the customers with unusually high calling times especially at night time. In the visualization presented in Figure 5, we therefore focus on customers with a high call volume between 2 a.m.and 3 a.m. (subwindow 4). Colors are chosen such that high call volumes correspond to high bright colors (yellow) and dark colors correspond to small call volumes). It is interesting that - for customers with a high call volume between 2 a.m. and 3 a.m. - there is a high inverse correlation between the time bin 2 a.m. - 3 a.m. and the time bins 6 a.m. - 7 p.m. Similarly, other interesting subsets of the database may be identified, helping to better understand the data and to identify characteristics of fraudulent calling behavior.

We also used the query-dependent techniques to analyze data obtained from AT&T´s internet service WorldNet (cf. Figure 6). The data analyzed is information about the internet

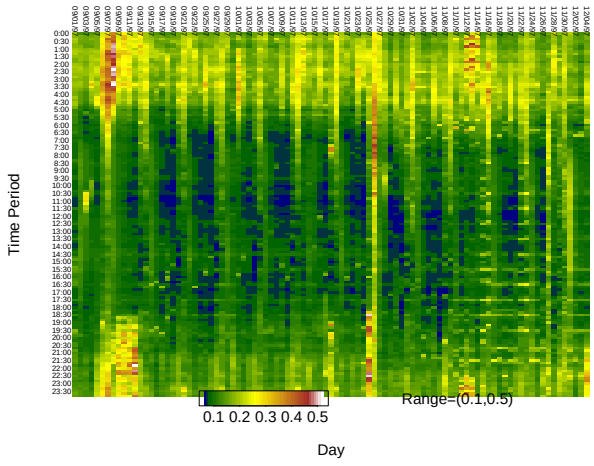

Figure 7: Recursive Pattern Visualization of Calls with a Specific Final Handling Code

4. The visualizations do not necessarily show the real data. In some cases, the data had to be modified to protect proprietary information.

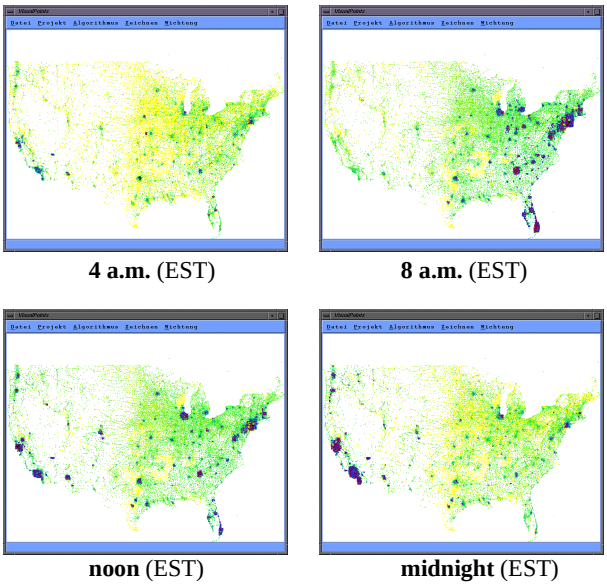

Figure 8: Time Series of Pixel-oriented Visualizations of Call Volume Data

sessions such as number of sessions per month, average length of the sessions, maximum length of the sessions, number of transferred bytes, etc. The focus of the query-dependent visualization is on customers with a high number of sessions and a high average length of the sessions (subwindow 3 and 5). It is interesting that there is a high correlation between those two attributes but the visualization also reveals a high correlation between the attributes shown in subwindows 6 to 9.

To apply the *query-independent techniques* it is important to have an natural ordering of the data. We used the recursive pattern visualization technique to look at calls with a specific final handling code in a certain region over a certain time period. The data is arranged such that one day corresponds to one column. The resulting recursive pattern visualization shown in Figure 7 clearly shows the daily time-dependency of the calls but it also reveals special time periods with an unusual behavior. The unusual behavior with a very low or very high number of calls the given final handling code (for example, Sept. 07-08, 1997 and Oct. 26, 1997) may either result from external events such as holidays, promotions, and catastrophes or internal events such as network failures.

The *time- and geometry-related pixel-oriented 2D displays* have been applied to time series of call detail data. Figure 8 presents four time steps of such visualizations, showing the call volume within a 10 minute interval at the given time. The time sequence clearly shows the development of the call volume over time. The visualization allows an intuitive understanding of the development of the call volume, showing the wake-up from east to west, the drop down in call volume at commuting and lunch time, etc. These aspects are expected patterns but more interesting are unexpected patterns which can easily be discovered in monitoring the call volume.

An interesting example of applying the *SWIFT-3D* system is the examination of calls that cannot be completed due to congestion at the customer premise. Keeping this number

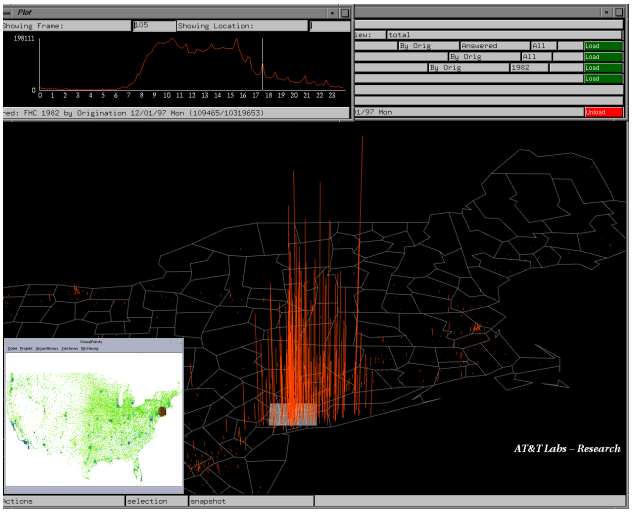

Figure 9: Inspection of Network Event's Effect on Customer

low is important due to the resources consumed. This is important both to the customers (who need reliable service for tele-marketing 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 9). 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.

5. 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.

Acknowledgments

The authors thank David Belanger, Ken Church and Manolis Tsangaris for their collaboration on network visualization, and especially with help to obtain experimental data. The authors also thank Simon Byers for providing the screen shot shown in Figure 7.

References

[AC 91] Alpern B., Carter L.: *'Hyperbox'*, Proc. Visualization '91, San Diego, CA, 1991, pp. 133-139.

[ADLP 95] Anupam V., Dar S., Leibfried T., Petajan E.: *'DataSpace: 3-D Visualization of Large Databases'*, Proc. Int. Symp. on Information Visualization, Atlanta, GA, 1995, pp. 82-88.

[AKK 96] Ankerst M., Keim D. A., Kriegel H.-P.: *'Circle Segments: A Technique for Visually Exploring Large Multidimensional Data Set'*, Proc. Visualization '96, San Francisco, CA, 1996.

[And 72] Andrews D. F.: *'Plots of High-Dimensional Data'*, in: Biometrics, Vol. 29, 1972, pp. 125-136.

[AS 82] Apperley M., Spence I. T.: *'A Bifocal Display Technique for Data Presentation'*, Proc. Eurographics, 1982, pp. 27-43.

[AS 94] Ahlberg C., Shneiderman B.: *'Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays'*, Proc. Human Factors in Computing Systems CHI '94 Conf., Boston, MA, 1994, pp. 313-317.

[Asi 85] Asimov D.: *'The Grand Tour: A Tool For Viewing Multidimensional Data'*, SIAM Journal of Science & Stat. Comp., Vol. 6, 1985, pp. 128-143.

[AW 95] Ahlberg C., Wistrand E.: *'IVEE: An Information Visualization and Exploration Environment'*, Proc. Int. Symp. on Information Visualization, Atlanta, GA, 1995, pp. 66-73.

[BSC 96] Buja A., Swayne D. F., Cook D.: *'Interactive High-Dimensional Data Visualization'*, Journal of Computational and Graphical Statistics, Vol. 5, No. 1, 1996, pp. 78-99.

[Bed 90] Beddow J.: *'Shape Coding of Multidimensional Data on a Mircocomputer Display'*, Visualization '90, San Francisco, CA, 1990, pp. 238-246.

[Bed 94] Bederson B.: *'Pad++: Advances in Multiscale Interfaces'*, Proc. Human Factors in Computing Systems CHI '94 Conf., Boston, MA, 1994, p. 315.

[BEW 95] Becker, R. A., EickS. G. and Wilks A.: *'Visualizing network data'*, IEEE Transactions on Visualization and Computer Graphics, 1(1), pp. 16-28, March 1995.

[BETT 99] Battista G. D., Eades P., Tamassia R., Tollis I.: *'Graph Drawin: Algorithms for the Visualization of Graphs'*, Prentice Hall, 1999.

[BEW 95] Becker R. A., Eick S. G., Wilks A. R.: *'Visualizing Network Data'*, Transactions on Visualization and Computer Graphics, Vol. 1, No. 1, 1995, pp. 16-28.

[BF 93] Beshers C., Feiner S.: *'AutoVisual: Rule-Based Design of Interactive Multivariate Visualizations'*, IEEE Computer Graphics and Applications, Vol. 13, No. 4, 1993, pp. 41-49.

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

[Cle 93] Cleveland W. S.: *'Visualizing Data'*, AT&T Bell Laboratories, Murray Hill, NJ, Hobart Press, Summit NJ, 1993.

[Eic 94] Eick S. G.: *'Data Visualization Sliders'*, Proc. ACM UIST, 1994, pp. 119-120.

[FB 94] Furnas G. W., Buja A.: *'Prosections Views: Dimensional Inference through Sections and Projections'*, Journal of Computational and Graphical Statistics, Vol. 3, No. 4, 1994, pp. 323-353.

[FS 95] Fishkin K., Stone M. C.: *'Enhanced Dynamic Queries via Movable Filters'*, Proc. Human Factors in Computing Systems CHI '95 Conf., Denver, CO, 1995, pp. 415-420.

[Fur 86] Furnas G.: *'Generalized Fisheye Views'*, Proc. Human Factors in Computing Systems CHI'86 Conf., Boston, MA, 1986, pp. 18-23.

[Hil 91] Hilbert D.: *'Über stetige Abbildung einer Linie auf ein Flächenstück'*, Math. Annalen, Vol. 38, pp. 459-460, 1891.

[HL 92] Herman G.T., Levkowitz H.: *'Color Scales for Image Data'*, Computer Graphics and Applications, 1992, pp. 72-80.

[Ins 85] Inselberg A.: *'The Plane with Parallel Coordinates, Special Issue on Computational Geometry'*, The Visual Computer, Vol. 1, 1985, pp. 69-97.

[ID 90] Inselberg A., Dimsdale B.: *'Parallel Coordinates: A Tool for Visualizing Multi-Dimensional Geometry'*, Visualization '90, San Francisco, CA, 1990, pp. 361-370.

[Joh 93] Johnson B.: *'Visualizing Hierarchical and Categorical Data'*, Ph.D. Thesis, Department of Computer Science, University of Maryland, 1993.

[Kei 96] Keim D. A.: *'Pixel-oriented Visualization Techniques for Exploring Very Large Databases'*, Journal of Computational and Graphical Statistics, Vol. 5, No. 1, 1996, pp. 58-77.

[Kei 97a] Keim D. A.: *'Visual Database Exploration'*, Tutorial, Int. Conf. on Knowledge Discovery in Databases (KDD '97), San Diego, 1997.

[Kei 97b] Keim D. A.: *'Visual Data Mining'*, Tutorial, Conf. on Very Large Databases, Athens, Greece, 1997.

[KH 98] Keim, D. A. and Herrmann A.: *'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.

[KK 94] Keim D. A., Kriegel H.-P.: *'VisDB: Database Exploration using Multidimensional Visualization'*, Computer Graphics & Applications, Sept. 1994, pp. 40-49.

[KK 95a] Keim D. A., Kriegel H.-P.: *'VisDB: A System for Visualizing Large Databases'*, System Demonstration, Proc. ACM SIGMOD Int. Conf. on Management of Data, San Jose, CA, 1995, p. 482.

[KK 95b] Keim D. A., Kriegel H.-P.: *'Issues in Visualizing Large Databases'*, Proc. Conf. on Visual Database Systems (VDB'95), Lausanne, Schweiz, März 1995, in: Visual Database Systems, Chapman & Hall Ltd., pp. 203-214, 1995.

[KKA 95] Keim D. A., Kriegel H.-P., Ankerst M.: *'Recursive Pattern: A Technique for Visualizing Very Large Amounts of Data'*, Proc. Visualization '95, Atlanta, GA, 1995, pp. 279-286.

[KKS 94] Keim D. A., Kriegel H.-P., Seidl T.: *'Supporting Data Mining of Large Databases by Visual Feedback Queries'*, Proc. 10th Int. Conf. on Data Engineering, Houston, TX, 1994, pp. 302-313.

[KK 96] Keim D. A., Kriegel H.-P.: 'Visualization Techniques for Mining Large Databases: A Comparison', Transactions on Knowledge and Data Engineering, Vol. 8, No. 6, Dec. 1996, pp. 923-938.

[KNTK 99] Koutsofios E. E., North S. C., Truscott R. and Keim D. A.: *'Visualizing Large-Scale Telecommunication Net-*

works and Services', Proc. Visualization '99, San Francisco, 1999.

[KNK 99] Koutsofios E. E., North S. C. and Keim D. A.: '*Visual Exploration of Large Telecommunication Data Sets'*, Visualization Blackboard, IEEE Computer Graphics and Applications, May 1999.

[LA 94] Leung Y., Apperley M.: *'A Review and Taxonomy of Distortion-oriented Presentation Techniques'*, Proc. Human Factors in Computing Systems CHI '94 Conf., Boston, MA, 1994, pp. 126-160.

[Lev 91] Levkowitz H.: *'Color icons: Merging color and texture perception for integrated visualization of multiple parameters'*, In Visualization '91, San Diego, CA, October 22-25 1991.

[LR 94] Lamping J., Rao R.: *'Laying out and Visualizing Large Trees Using a Hyperbolic Space'*, Proc. UIST, 1994, pp. 13-14.

[LRP 95] Lamping J., Rao R., Pirolli P.: *'A Focus + Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies'*, Proc. Human Factors in Computing Systems CHI '95 Conf., Denver, CO, 1995, pp. 401-408.

[LWW 90] LeBlanc J., Ward M. O., Wittels N.: *'Exploring N-Dimensional Databases'*, Visualization '90, San Francisco, CA, 1990, pp. 230-239.

[MB 95] Munzner T., Burchard P.: *'Visualizing the Structure of the World Wide Web in 3D Hyperbolic Space'*, Proc. VRML '95 Symp, San Diego, CA, 1995, pp. 33-38.

[Mor 66] Morton G. M.: *'A Computer Oriented Geodetic Data Base and a New Technique in File Sequencing'*, IBM Ltd. Ottawa, Canada, 1966.

[MRC 91] Mackinlay J. D., Robertson G. G., Card S. K.: *'The Perspective Wall: Detail and Context Smoothly Integrated'*, Proc. Human Factors in Computing Systems CHI '91 Conf., New Orleans, LA, 1991, pp. 173-179.

[Pea 90] Peano G.: *'Sur une courbe qui remplit toute une aire plaine'*, Math. Annalen, Vol. 36, pp. 157-160, 1890.

[PG 88] Pickett R. M., Grinstein G. G.: *'Iconographic Displays for Visualizing Multidimensional Data'*, Proc. IEEE Conf. on Systems, Man and Cybernetics, IEEE Press, Piscataway, NJ, 1988, pp. 514-519.

[RC 94] Rao R., Card S. K.: *'The Table Lens: Merging Graphical and Symbolic Representation in an Interactive Focus+Context Visualization for Tabular Information'*, Proc. Human Factors in Computing Systems CHI '94 Conf., Boston, MA, 1994, pp. 318-322.

[RMC 91] Robertson G. G., Mackinlay J. D., Card S. K.: *'Cone Trees: Animated 3D Visualizations of Hierarchical Information'*, Proc. Human Factors in Computing Systems CHI '91 Conf., New Orleans, LA, 1991, pp. 189-194.

[SB 94] Sarkar M., Brown M.: *'Graphical Fisheye Views'*, Communications of the ACM, Vol. 37, No. 12, 1994, pp. 73-84.

[Shn 92] Shneiderman B.: *'Tree Visualization with Treemaps: A 2D Space-Filling Approach'*, ACM Transactions on Graphics, Vol. 11, No. 1, pp. 92-99, 1992.

[STDS 95] Spence R., Tweedie L., Dawkes H., Su H.: *'Visualization for Functional Design'*, Proc. Int. Symp. on Information Visualization (InfoVis '95), Atlanta, GA, 1995, pp. 4-10.

[War 94] Ward M. O.: *'XmdvTool: Integrating Multiple Methods for Visualizing Multivariate Data'*, Visualization'94, Washington, DC, 1994, pp. 326-336.

[WL 93] van Wijk J. J., van Liere R.. D.: *'Hyperslice'*, Proc. Visualization '93, San Jose, CA, 1993, pp. 119-125.

[Wri 95] Wright W.: *'Information Animation Applications in the Capital Markets'*, Proc. Int. Symp. on Information Visualization, Atlanta, GA, 1995, pp. 19-25.

[WUT 95] Wilhelm A., Unwin A.R., Theus M.: *'Software for Interactive Statistical Graphics - A Review'*, Proc. Int. Softstat '95 Conf., Heidelberg, Germany, 1995.