A Java-based Visual Mining Infrastructure and Applications

Ming Hao, Umesh Dayal, Meichun Hsu, Jim Baker*, Robert D'Eletto*
Software Technology Laboratory
HP Laboratories Palo Alto
HPL-1999-49
April, 1999

E-mail: [mhao,dayal,mhsu]@hpl.hp.com
jim_baker@hp-usa-om14.om.hp.com
bob_deletto@hp-usa-om13.hp.com

visual mining,
Java-based,
hidden structure
and relationship,
Knowledge Discovery

Many real-world KDD (Knowledge Discovery & Data Mining) applications involve the navigation of large volumes of information on the web, such as, Internet resources, hot topics, and telecom phone switches. Quite often users feel lost, confused, and overwhelmed with displays that contain too much information. This paper discusses a new content-driven visual mining infrastructure called VisMine, that uses several innovative techniques: (1) hidden visual structure and relationships for uncluttering displays; (2) simultaneous visual presentations for high-dimensional knowledge discovery; and (3) a new visual interface to plug in existing graphic toolkits for expanding its use in a wide variety of visual applications. We have applied this infrastructure to three data mining visualization applications – topic hierarchy for document navigation, web-based trouble shooting, and telecom switch mining.

Internal Accession Date Only
*Hewlett-Packard Company, Englewood, CO

A Java-based Visual Mining Infrastructure and Applications

Ming C. Hao, Umesh Dayal, Meichun Hsu, Jim Baker, Robert D'Eletto
(mhao, dayal, mhsu)@hpl.hp.com
jim_baker@hp-usa-om14.om.hp.com
bob_deletto@hp-usa-om13.hp.com
Hewlett Packard Research Laboratories

Abstract

Many real-world KDD (Knowledge Discovery & Data Mining) applications involve the navigation of large volumes of information on the web, such as, Internet resources, hot topics, and telecom phone switches. Quite often users feel lost, confused, and overwhelmed with displays that contain too much information. This paper discusses a new content-driven visual mining infrastructure called VisMine, that uses several innovative techniques: (1) hidden visual structure and relationships for uncluttering displays; (2) simultaneous visual presentations for high-dimensional knowledge discovery; and (3) a new visual interface to plug in existing graphic toolkits for expanding its use in a wide variety of visual applications. We have applied this infrastructure to three data mining visualization applications – topic hierarchy for document navigation, web-based trouble shooting, and telecom switch mining.

Keywords: Visual Mining, Java-Based, Hidden Structure and Relationship, Knowledge Discovery

1. 0 Introduction

Recently, the fast growth of information and the Internet have led to the availability of large volumes of data. Recent research efforts have focused on visual mining in many different areas, such as telecom switch data, World Wide Web traffic, company organization charts, and file systems [1,2,6,7,8,13]. In industry, IBM's Intelligent Miner organizes data so as to make maximum use of significant pattern recognition. SGI's MineSet uses 3D animation to represent knowledge extracted from large data sets. AT&T Bell Laboratory's SeeNet uses 3D layout and direct user graphic interfaces to visualize telecom network activities. MindMan [4] helps individuals organize, generate and learn ideas and information with multiple displays.

Next generation information visualization systems will be very different from those of today. Current information visualization systems are designed to handle moderate amounts of structured data. New information visualization systems will be built around the navigation of, and interaction with, massive volumes of unstructured information. Many major issues need to be addressed, especially, those involving new techniques for mining knowledge from large data warehouses. For example, a telephone company needs to analyze millions of call records to decide whether it needs to add another tandem switch. A customer service center needs to distill a solution to a customer's problem. Companies need to access data by market segment in order to forecast business trends. The challenge is to find methods for presenting valuable information from large volumes of data so as to enable a user to quickly identify exceptions and to distinguish interesting patterns visually.

The following are some issues in today's visual mining of massive volumes of data:

1. cluttered display
2. disjoint displays
   (display after display presentations)
3. limited access & lack of expandability

Information visualization based on a single complex view often causes display clutter and visual confusion. Besides, single view visualization does not allow users to visualize the inter-relationships among different sets of high-dimensional datasets. A common solution to provide multiple views is to use many displays. But users have to click through display after display to find the information. For example, in a telecom switch mining application, suppose a user wants to selectively monitor overloaded telephone links in the United States. Starting with a display of a United States map, the user would need to click though each display of progressively greater detail (at the state level,

at the city level, etc) until the user finds the overloaded links. With multiple views, the user can see presentations at different levels of detail simultaneously to identify the problem real time. Often, users have great difficulty to analyze and correlate information from these disjoint displays.

Most recent graphic toolkits are designed for visualization of certain types of knowledge. Inxight's hyperbolic tree toolkit [5] is designed for representing hierarchical relationship knowledge, not for unstructured data, such as geographical telecom switch visualization. To meet different requirements, we need a new visual interface to plug-in multiple graphic toolkits.

2.0 Our Approach

At HP Laboratories, we have devised some visualization solutions to resolve the above difficulties. One method is to hide visual structure and relationships to reduce display cluttering and visual confusion. This method hides all non-primary relationships; it only shows objects when the user focuses on them. All other structures and relationships are hidden in the property of each object. Another method is to directly interact with the user and mining engines to slice and dice large complex knowledge into multiple simultaneous presentations. This method allows a user to easily discover knowledge relationships and exceptions. The third method is to define new visual interfaces to plug into existing graphic toolkits, such as TGS' 3DMSJava [3] and Inxight's Hyperbolic Tree Toolkits, thus expanding the use of our visualization infrastructure to a wide variety of visual applications. These methods are driven by information content. The technology to encompass all these methods is referred to as VisMine (Visual Mining).

We have applied this infrastructure to three data mining visualization applications – topic hierarchy for document navigation, web-based trouble shooting, and telecom switch mining. For example, a topic hierarchy consists of a tree structure representing the primary topic-subtopic relationship plus many cross links representing other secondary relationships among topics. VisMine hides the structures and relationships that are not currently in focus. VisMine applies the same method of information hiding to help a user troubleshoot a highly interconnected problem on the web. In the telecom switch mining application, VisMine hides sub-region traffic distribution in the region property instead of displaying detailed structures. For rapid discovery of patterns, VisMine is able to monitor multiple simultaneous views of telecom traffic. The user does not need to click through display after display to find the information needed. VisMine plugs together various existing toolkits to meet different needs. For example, VisMine uses Inxight's hyperbolic tree toolkits for navigating a topic hierarchy and for web-based trouble shooting. VisMine uses TGS 3DMSJava for monitoring telecom switch traffic.

3.0 Component Architecture

VisMine is built on a Java-based client-server model. Java allows the creation of visual component applets that can be automatically downloaded and executed on the local client. To achieve rapid display, VisMine separates the visualization from the data mining computation engine. The data mining computation executes on the server. The visualization construction and rendering are done locally in the client sites. As a result, VisMine is able to provide fast response needed for real-time visual mining.

VisMine architecture [9,14] contains three basic components:

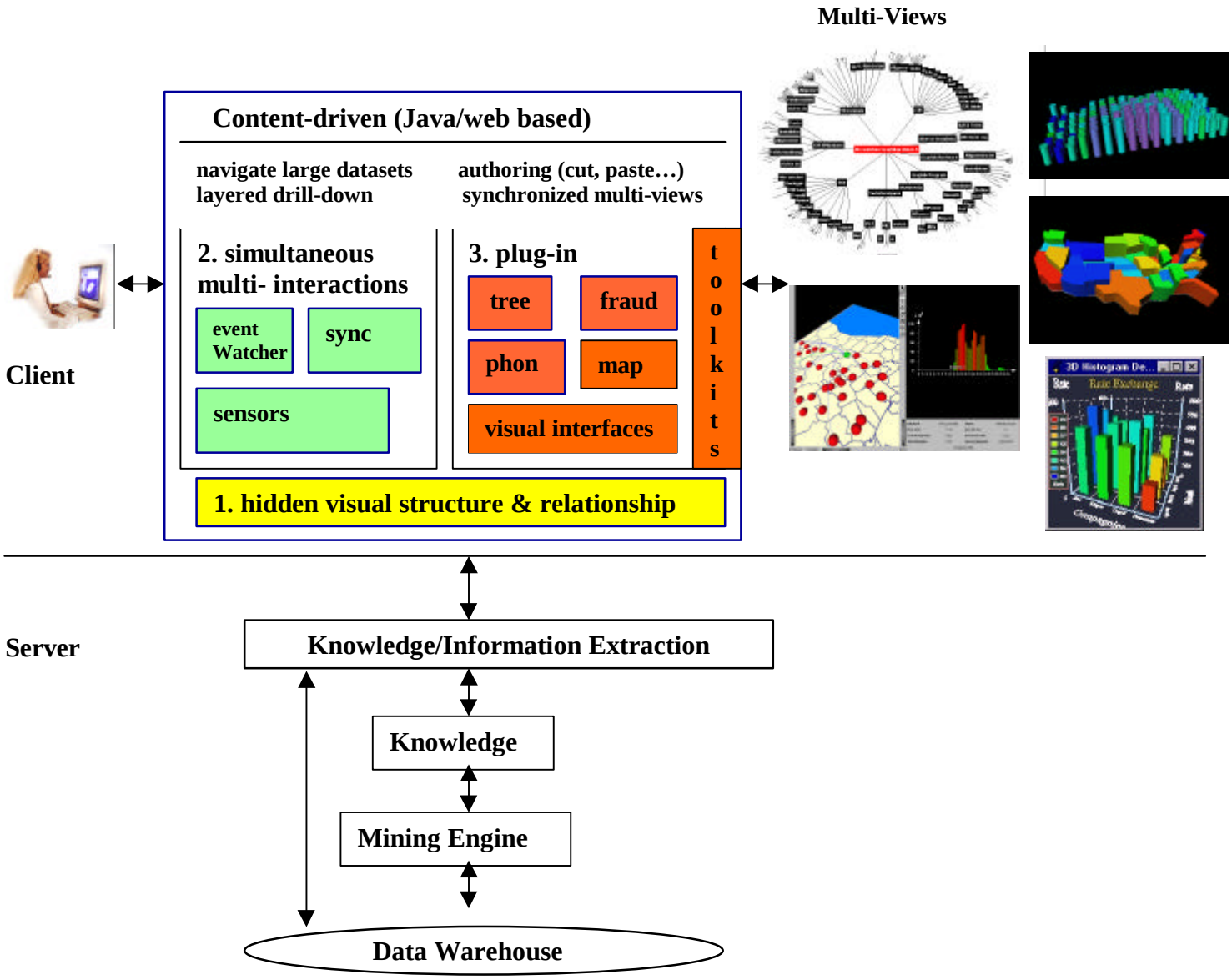
1. hidden visual structure and relationships
2. simultaneous multiple presentations
3. plug-in capability (visual interfaces)

Figure 1 illustrates the VisMine overall architecture. Each of the above components is described further in the following sections.

To make large high-dimensional knowledge easy to mine and interpret, VisMine emphasizes the following capabilities:

- navigate and author large knowledge bases
- layered drill-down
- synchronized multi-representations
- multi-dimensional presentations with 2D/3D color, zoom, and rotate.

Figure 1: VisMine Component Architecture



3.1 Hidden Visual Structure & Relationships

A common method for visualizing an application, such as fault diagnosis, document topic hierarchy navigation, or market segment analysis is to layout all the structure and relationships on the screen, such as a spatial layout with nodes and links, or a matrix layout with cells. However, for large complex information, these static techniques do not work. There are too many lines, nodes, and cells to draw. As a result, the display becomes cluttered and causes visual confusion.

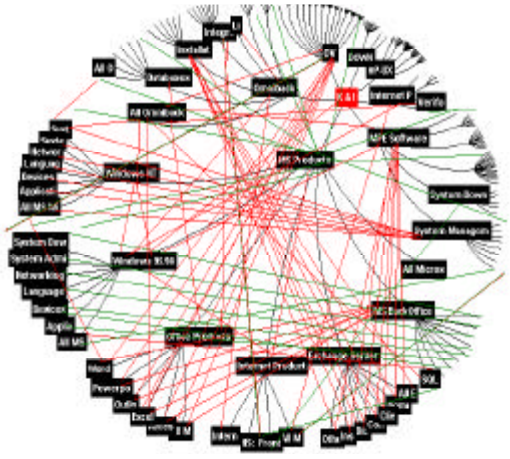
VisMine employs a visual hierarchy tree to map the knowledge for automatic adjustment of complexity. VisMine organizes the visual hierarchy tree into different levels of group and display sequences according to each application categorical feature and user input parameter. For example, the visual tree can be organized

according to the hierarchical relationship in a topic hierarchy or corporate organization chart application. Also, the visual tree can be organized according to a geographical relationship or service categories for network applications.

This technique hides all the non-primary structure and paths in each object's property at the time the visual space is initialized. They become accessible and interactive only at the time of focus. The user can easily navigate through all possible paths without tracing many lines and intersections. For example, in an organization chart, a user may have a primary manager and several temporary mangers. When a user clicks on an object that has a hidden link to his or her temporary manager in addition to a visible link to his or her primary manager, this technique automatically maps the path to the temporary manager structure to facilitate navigation. At the conclusion of navigation, VisMine removes the path to the temporary manager. Figure 2 illustrates the difficulties of visualizing a graph with multiple paths (non-primary, cross-link) without using this hidden structure method.

Figure 2: A Multi-path Hyperbolic Space with and without Hidden Links

(A) Traditional method: No hidden link



This multi-path hyperbolic graph becomes very cluttered. There are many lines, intersections, and broken lines.

The display cluttering is caused by drawing lines to connect to the related nodes.

(B) Our Method: With hidden links

This multi-path hyperbolic space retains the simplicity of the original hyperbolic space without using lines to link to the related nodes. This hidden link technique hides non-primary paths in each object's property. They become accessible and interactive only at the time of focus. The user can easily navigate through all possible paths without tracing many lines and intersections.

For example, the "Office Product" node contains a hidden-path to "MS BackOffice" (indicated as an empty circle). No extra line is needed to connect MS BackOffice node to the Office Product node

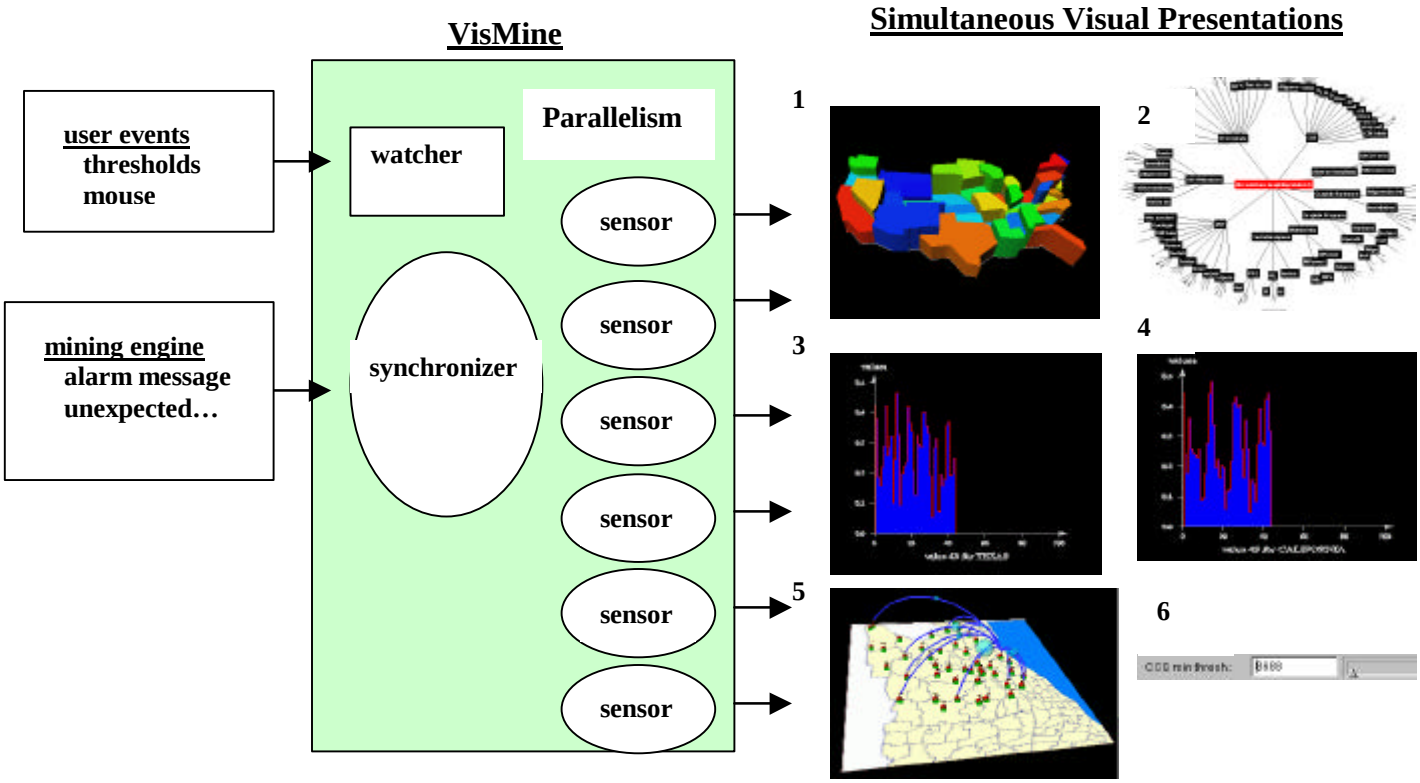
The "MS BackOffice" child nodes will be automatically mapped from its parent node-"MS Product" to the "Office Product", when the user clicks on the "Office Product".

3.2 Simultaneous Multiple Presentations

There are two types of event interactions. One is direct from the Web user interface. For example, the user moves the slider on the display to select phone service units to visualize. The other type of event is from the mining engines. For example, an unexpected condition happened in the middle of mining telecom switch traffic data. VisMine assigns each visual group a sensor. The sensor will be automatically awakened to handle these incoming events and execute certain functions, such as setting up an alarm or activating a spotlight.

For visual consistency, a change in one presentation will be propagated in real-time to the other presentations. VisMine employs a synchronizer to ensure that all changes occur simultaneously. For example, a user can set a new threshold during visualization, and the results will be reflected in all the presentations synchronously. Figure 3 illustrates animated telephone traffic data in the United States. The six views are synchronously presented at the same time. Any change in one view will be automatically made in the other five views simultaneously. The Texas and California histograms are presented simultaneously to describe the 24-hour telephone traffic. This capability enables a user to visually detect the traffic patterns between Texas and California and their relationships with the rest of the country.

Figure 3: Simultaneous Multiple Presentations

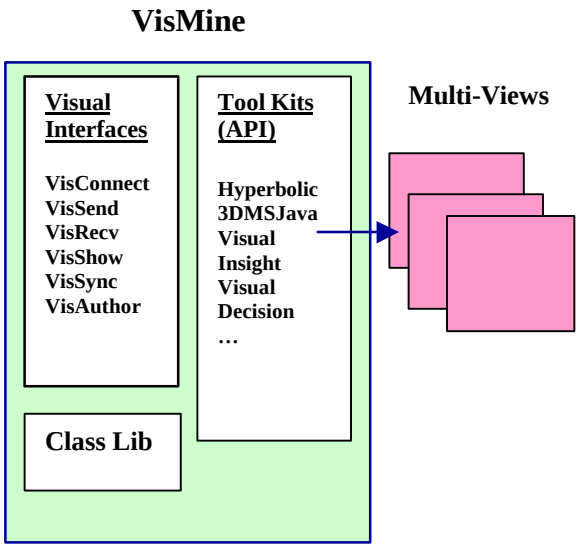


Presentation 1: use a United States map to visual national-wide telephone traffic.
Presentation 2: use a hyperbolic tree to represent the logical telecom switch connections.
Presentation 3 & Presentation 4: use two histograms to represent Texas and California traffic data.
Presentation 5: use animated presentation of 24-hours phone traffic in San Francisco.
Presentation 6: use a text field to enter customer service seconds to visual different traffic patterns.

3.3 Plug-in Capability

Different types of application require different visual representations. Sometimes, even within the same application, there is a need to use different graphic techniques for different visual representations. For example, in telecom switch management, we need to have a switch hyperbolic tree layout for service classification in a 3D geographical representation. VisMine defines a visual interface to plug in various existing toolkits. For example, VisMine interfaces with Inxight's hyperbolic tree toolkits to represent the service classification hierarchy, and it interfaces with TGS's toolkits to construct 3D-network visualizations for displaying maps and histograms of traffic service data.

Figure 4: Visual Interfaces & Plug-In



To interface to different existing toolkits, (as shown in Figure 4) VisMine defines a common visual interface, a class library, and a suite of visual protocols, such VisConnect, VisSend, VisRecv, VisSync, VisAuthor as the following:

- VisConnect: to connect to an existing toolkit.
- VisSend: to send the current visual data and message to the existing toolkit.
- VisRecv: to receive the current data and message from the existing toolkit.
- VisShow: to inform the graphic toolkit to display the view
- VisSync: to synchronize multiple views for simultaneous presentations
- VisAuthor: to perform knowledge tree authoring (cut, paste, add, …)

The VisMine class library contains visual interfaces to the existing graphic toolkits API.

4.0 Web Data Mining Applications

VisMine uses a web browser with a Java activator to allow real-time interactive visual mining on the web, as illustrated in Figure 5. The Web interfaces are based on standard HTML and the use of Java applets, which are used to explore relationships and to retrieve data within a region of interest. The server is integrated with the data warehouse and mining engine. The user at the client side visually mines the knowledge results. It allows the user to dynamically access large hierarchies with complex links through HTML pages in a Web browser.

There are many data mining applications with large information structures that can employ the VisMine content-driven execution model. We have prototyped several applications, which we will use to illustrate these techniques.

Figure 5: VisMine Client-Server Web Structure

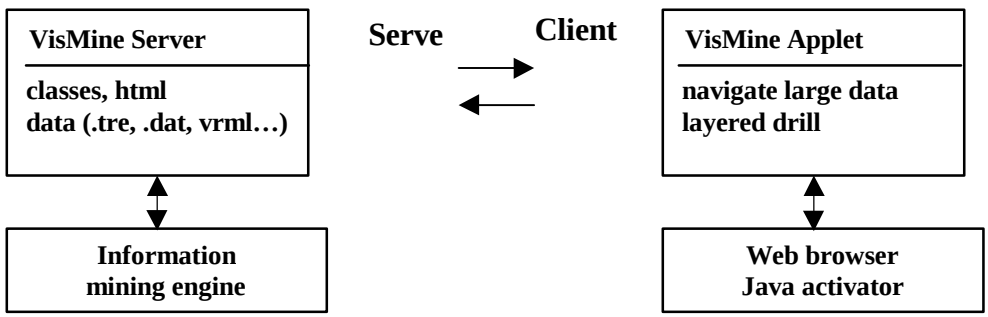


Figure 6: A Web Topic & Content Hierarchies Sample

This graph represents a topic hierarchy of 200,000 documents on the web.

The user can navigate the tree and read a document on demand.

There are many hidden non-hierarchical relationships among nodes. These relationships are only constructed and displayed when the user focuses on them.

4.1 Topic Hierarchy for Document Navigation

The first example is to visualize a topic hierarchy for document navigation [11]. A hyperbolic space is constructed to present a topic hierarchy for millions of documents linked to the web. The topic hierarchy is constructed by mining the content of the documents and session logs that record accesses to these documents. Using the hidden structure and relationships capabilities, we are able to navigate a large, highly connected topic hierarchy in a simple, uncluttered hyperbolic space on a web screen. (As shown in Figure 6)

4.2 Web-Based Trouble Shooting

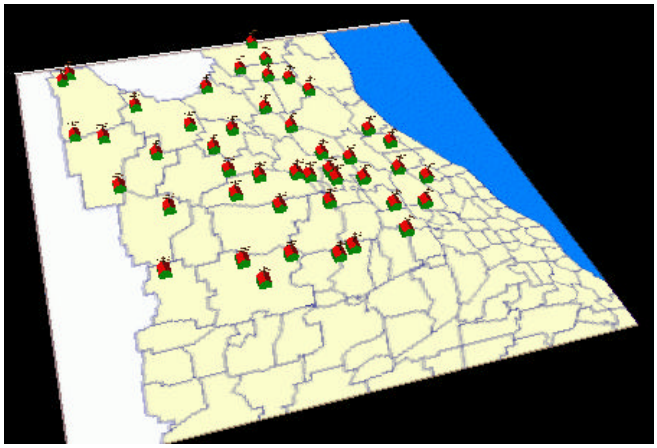
The second example is a Hewlett Packard internal web-based troubleshooting application called InterView [10]. We use a hyperbolic space to organize and view large numbers of questions and answers in a hierarchical structure. Questions are parent nodes, while answers are arcs to child nodes. A question can have several answers. An answer can lead to another set of questions and answers. Using the hidden structure and relationship capabilities, users are able to navigate through answers to link to another question and answer set that does not directly belong to the primary path of the hyperbolic space. The hidden structures and relationships give the hyperbolic space the ability to represent a general network, and thus are not restricted to parent-child structured trees. A user can easily follow the knowledge content to search for recent patches, technical tips, and versions. In addition, statistical data can be mapped to the hyperbolic space using colored nodes or arcs.

4.3 Telecom Switch Mining

The third example is visual mining of telecom network switch data. VisMine organizes the geographical location of a phone switch and the monitoring policy into different group levels, such as country, state, service units, and busy hours for mining visually. VisMine provides buttons, sliders, color scalar, threshold, and other visual metaphors to permit a user to dynamically choose the region of interest. VisMine hides non-primary structures and relationships and only displays objects when in focus. For example, VisMine hides all the phone switches located in San Francisco from the United States map. Upon a user's request, VisMine is able to show all phone activity with service units above 2,000 call seconds in San Francisco. Using 3D animation, VisMine is able to show changes of calling behavior. By selecting a proper visual level, a user can easily identify which switch links (Trunks) are being overloaded indicating that a second switch may need to be added to balance the load.

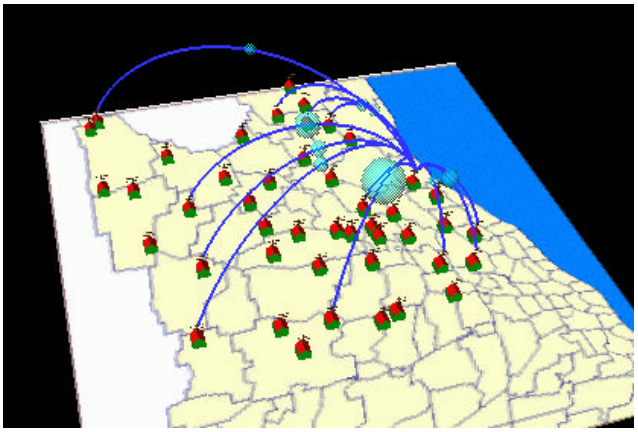
Figure 7, Figure 8, and Figure 9 illustrate a 24-hour period of phone switch service with and without animation. To visually mine the traffic, VisMine uses: (1) a house with a pole to represent a phone switch location; (2) an arc to represent the link between the caller and the callee; (3) differently sized spheres to represent the number of service units; (4) an arrow to represent the calling direction; (5) a cylinder to represent the link in real-time; (6) time series to represent the time changes within 24 hours; (7) an encoded color chart to represent the number of calls (8) a slider to control the threshold for visually mining different switch levels and groups.

Figure 7: A 24 Hours Telecom Phone Service



This city map is displayed as a result of clicking on San Francisco in the United States Map.

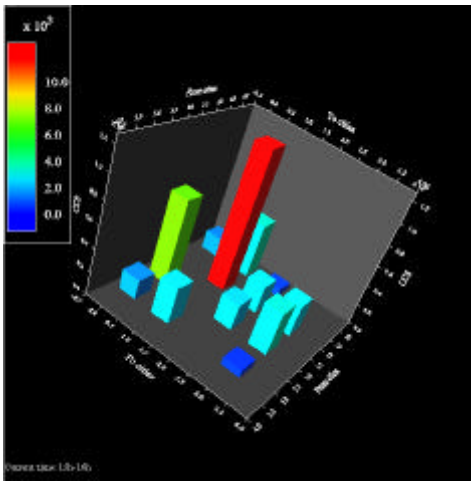
This map shows:
houses and poles to represent the geographical distribution of switches**.**



(Focus on houses and links to monitor the current telecom traffic:

- discover one switch overloaded (big sphere)
- identify the heavy caller and callee (house) and the link(arc)
- click on the house to display a 24 hours of telecom phone traffic (shown in Figure 9B**)**

Figure 8 An Animated 24 Hour Real-time Multi-Dimensional Cube



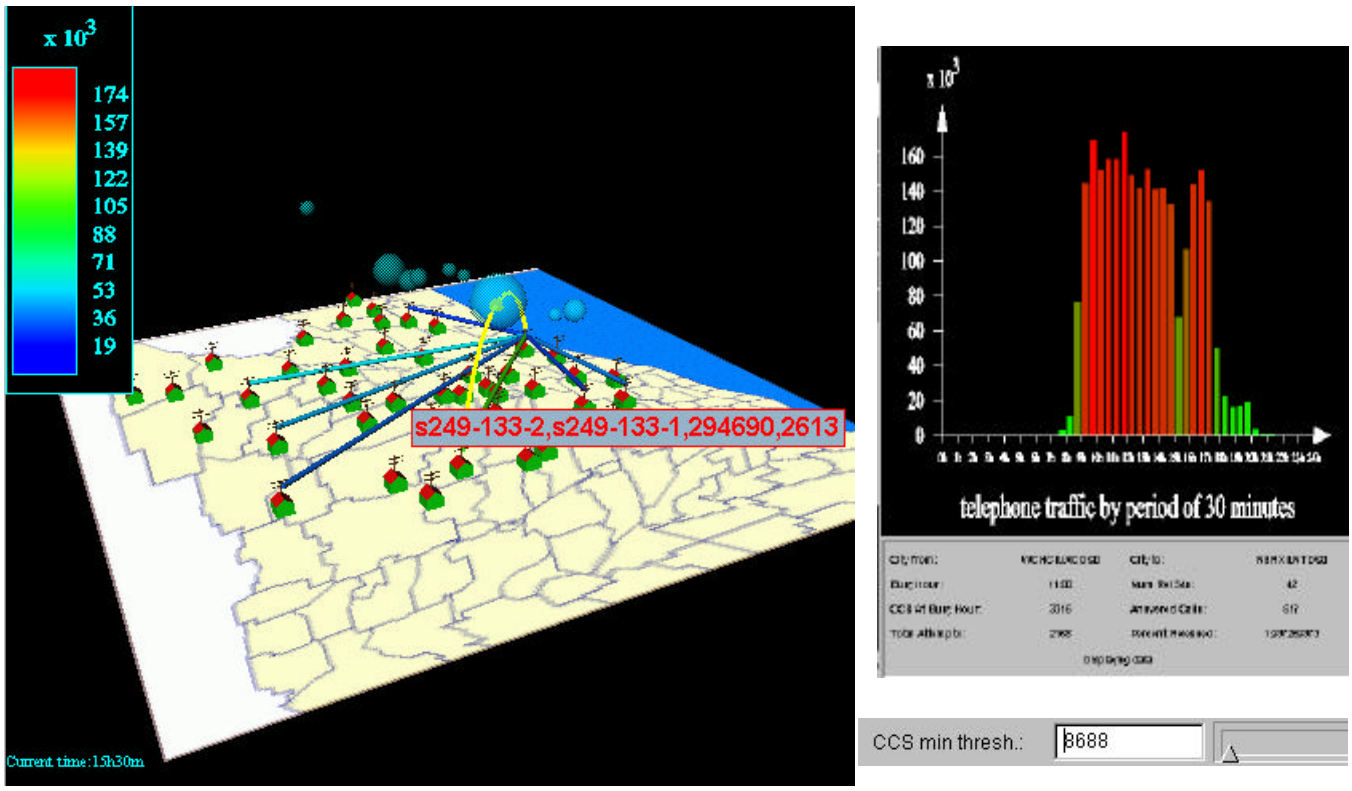
Use a 3D cube to represent multi-dimensional real-time presentation.

- use x axis to represent callers
- use y axis to represent callees
- use z axis to represent service units (sec)
- use color to represent number of calls
- use time series to represent the changes

These 5 dimensions can be arbitrarily assigned to any relationships**.**

Figure 9: An Animated 24 Hour Real-time Telecom Phone Service

(A) Geographical Presentation **(B) Histogram**



From the animated geographical map: (information displayed only when in focus)

- a heavy-loaded switch service found on the link from s249-133-2 to s249-133-1
- current activity occurred at 11:30
- current threshold limit at 8688
- the amount service 294,690 sec
- total number of calls 2,613
- a phone traffic histogram contains detailed call data
- a service unit (call sec) slider for entering different threshold

5.0 Conclusion

Data mining applications face difficulties in the visual mining of massive, highly connected data sets on the Internet. To date, many practical applications have shown the usefulness of hyperbolic space [2, 3] and telecom data mining [7,12,13].

In this paper we describe a new visual mining infrastructure called VisMine. In VisMine, we define a new visual interface to enable the use of various existing visual toolkits. VisMine employs a hidden structure and relationship method to unclutter the display for massive data visualization. In addition, VisMine allows users to easily navigate through different links without being overwhelmed by a large number of nodes and paths. These techniques have been successfully prototyped at Hewlett Packard Laboratories.

6.0 Acknowledgements:

Thanks to Joe Sventek and Martin Griss from HP STL (Software Technology Laboratories) for their suggestions and encouragement. Also thanks to Graham Pollock from STL for review and comments. Patrick Barthelemy and Patrick Vigneras from "Template Graphics Software", Jeff Holmbeck, Hatold Shinsato from "Inxight Software, Inc." for technical support and for allowing us to use their toolkits in our experiments.

7.0 References:

[1] Charlie Gunn, "Discrete Groups and Visualization of Three-dimensional Manifolds" ACM 1993.
[2] Tamara Munster, "Exploring Large Graphs in 3D Hyperbolic Space" IEEE Computer Graphics. Vol. 18, Number 4. 1998.
[3] Template Graphics Software, San Diego, CA.
[4] MindMan is a visualization and organizational tool to help individuals to organize, generate and learn ideas and information. 1998.
[5] The Hyperbolic Tree Toolkit is a product from Inxight Software for exploiting large amounts of information. 1998
[6] John Lamping and Ramana Rao, "Laying out and Visualizing Large Trees Using a Hyperbolic Space". ACM /UIST'94.
[7] Stephen G. Eich, "Aspects of Network Visualization", IEEE Computer Graphics and Applications, March 1996.
[8] Joe C. Pinheiro, Don X Sun, "Methods for Linking and Mining Massive Heterogeneous Databases, KDD98.
[9] Ming C. Hao, Meichun Hsu, Umesh Dayal, Adrian Krug, "A Technique for visualizing Large Web-based Hierarchical Hyperbolic Space with Multi-Paths", the Third International Conference on the Practical Application of Knowledge Discovery and Data Mining, PADD99, April 1999.
[10] Adrian Krug: "InterView: Knowledge Content & Program Architecture" Hewlett Packard SW-Support Delivery Engineering, 1998.
[11] Qiming Chen, Parvathi Chundi, Umesh Dayal, Mei Hsu, "Dynamic Agents for Dynamic Service Provisioning" Int. Conference. 1998.
[12] Daniel A. Keim, Annemarie Herrmann, "The Gridfit Algorithms: An Efficient and Effective Approach to Visualizing Large Amounts of Spatial Data", IEEE Visualization'98.
[13] Stephen G. Erick and Graham J. Wills, "Navigating large networks with hierarchies" IEEE Visualization '93.
[14] Ming C. Hao, Meichun Hsu, Umesh Dayal, "Method and Apparatus for Visual Mining of Multiple Simultaneous Presentations by Plugging in a Plurality of Existing Graphic Toolkits", Internal Technical Report, HEWLETT PACKARD Palo Alto Research Lab, 3/1999.