Web-Based Information Visualization

Randall M. Rohrer and Edward Swing
National Security Agency

Increasingly, the World Wide Web is being used to help visualize complex relational information. Here we discuss several Web-based visualization prototypes and applications.

Information visualization, an emerging discipline, uses visual means to represent nonspatial, abstract data. To visualize such information, you must map this data into a physical space. Finding the appropriate visual mapping for the task at hand proves vital to producing effective visualizations. Information visualization can often help you find and understand relationships and structure within (seemingly) unstructured data. Recent widespread interest has focused on exploration of information visualization techniques and applications for just that reason. At the same time, information has become pervasive thanks to underlying mechanisms such as the World Wide Web (WWW) and corporate intranets. Visualizing Web-based information—either from the WWW or intranets—has become a common application of information visualization.

Given these trends, the Web has naturally progressed as a source of information as well as an underlying delivery mechanism for interactive information visualization. To further explore these ideas, developers use tools such as Virtual Reality Modeling Language (VRML), Java, and Web browsers such as Netscape to create Web-based information visualization applications. While a Web-based delivery mechanism offers a number of advantages, it also imposes a number of limitations and problems.

A fundamentally new medium for visualization, the Web is changing the way visualization applications are developed, delivered, and used. We have developed a number of Web-based information visualization prototypes and applications by adapting several well-known information visualization ideas and techniques for use within Web environments. Before delving into specific examples, we offer some relevant background about the Web and our use of visualization for analysis.

A new medium for visualization

The Web provides a flexible means for linking applications, data, information, and users. To seamlessly interlink associated data and couple visual representations with this data creates opportunity for new approaches to visualization. We use the term Web-based information visualization to describe visualization applications that use the Web as an information source, a delivery mechanism for visualization, or both. In fact, the synergistic effect of integrating the source and delivery aspects of the Web provides strong support for visualization in this new medium.

A Web undercarriage affects how you approach visualization. The Web provides a loose coupling between data, users, and applications. While this gives flexibility for remote data access and logical associations between data, it also means limited knowledge transfer between client and server. Visualization applications targeted for the Web need to account for this. Current Web interaction and navigation techniques follow an intuitive point-and-click paradigm that lets users follow associated hyperlinks and drill down to underlying data. Web-based visualization interactions often follow a similar approach and may well evolve into new hyperlink-based interaction paradigms.

Web-based visualization lets users customize applications and data representations not originally targeted for each other by dynamically linking Web-based data and visualization applications. Push technologies, which send (push) information to a user's desktop, automatically deliver data and information to users through customized subscription to data services. This flood of information flowing to the desktop may require visualization for succinct presentation and structuring. To be effective, these pushed visualizations will need to integrate disparate pieces of information.

Web-based visualization also provides an easy mechanism for incorporating Web-based multimedia data into applications. The Web provides a rich, flexible new medium for information visualization applications. New techniques and paradigms will continue to emerge as we gain experience using the Web.

0272-1716/97/$10.00 © 1997 IEEE

Web-based development tools

In adapting visualization techniques for use in Web environments, we employ several common development tools, including VRML as a 3D graphics engine and standard Web interfaces such as Netscape for formatted text browsing. Currently, Hypertext Markup Language, or HTML, remains a common standard for providing formatted, interlinked text. Common Gateway Interface, or CGI, provides a method for interfacing Web tools with external gateway programs. These simple Web-based navigation and modeling tools permit using hyperlinks and CGI-triggered database queries to interlink related data and information. In addition, integrating Java with these tools enables the development of interactive, distributed Web applications.

VRML has become a popular de facto standard for producing 3D graphics within Web environments. Currently, a simple scripting language defines VRML scenes or object descriptions. Defined as a standard Multipurpose Internet Mail Extensions (MIME) data type, VRML scenes can be accessed through standard hypertext transfer protocol (HTTP) Web server mechanisms. VRML-capable browsers and plug-ins on the client side let you access and interact with VRML content that's published on a Web server. In addition to standard 3D graphics functionality, VRML embeds hyperlinks directly into 3D worlds. Individual objects can be hyperlinked to any Web media type such as text, images, movies, sounds, and other VRML worlds. These embedded hyperlinks can also trigger CGI-based actions such as database queries.

The initial version of VRML (VRML 1.0) defines a static scene. Visualizations based on VRML 1.0 prove somewhat limiting because users can only navigate and manipulate scenes as a whole and follow available hyperlinks. However, despite these limitations, a somewhat primitive interaction and dynamic update mechanism can be achieved by using server-side CGI programs for controlling access to data, processing, and broadcasting scene updates to VRML clients.

The updated VRML 2.0 standard adds the ability to incorporate dynamic behavior, animation, and user interaction with Java and JavaScript as an underlying computation engine on the client side. This improved version of VRML makes it easier to produce dynamic and interactive Web-based visualizations.

Java has become a popular mechanism for providing client-side interaction in Web applications. It provides a more tightly coupled communications and computation mechanism for Web-based applications, which typically have a loose coupling between client and server. Java interpreters—embedded within standard browsers such as Netscape—provide the underlying dynamics engine of VRML 2.0. To date, we have produced our Web-based visualizations by directly generating VRML scenes from an application program. In the future, we plan to adapt our work to higher level toolkits and Java-based solutions.

Visualizing Web-based information

You can visualize several structures inherent in Web-based information. These structures include Web topology, external structure, and cognitive models. Depending on the application goals and data set characteristics, these Web structures can play a significant role in choosing and designing visualization techniques.

The first inherent structure is the topology of the Web. This includes hyperlinks from page to page (media to media) as well as internal structure within documents. Typically, Web authors or content creators define these relationships and links. Traversing these links or topology remains the most common model of Web navigation and browsing.

The second inherent structure comes from structure imposed from outside the Web (external structure). External structure is often a logical grouping—for example, an organization's Web page hierarchy grouped and displayed by department or office. This may prove useful to identify the work of various offices as they contribute to the whole organization. Another example of external structure is the physical layout, geographical layout, or communications model of interconnected Web servers. Sometimes this external logical structure closely matches the underlying Web topology.

The cognitive model the user perceives comprises the third inherent structure. This structure depicts content or topic relationships that associate two or more pieces of information. These pieces of related information may come from separate topic domains and may or may not be explicitly linked to each other. Users frequently create their own desired cognitive mapping by grouping related information in a personalized structure such as a bookmark, hot list, or Web page. In fact, researchers are continually studying ways to help people track and organize these cognitive relationships.

One of the primary strengths of Web-based information systems is that they abstract and hide these various structures from users. For example, you do not need to know the geographic layout of interconnected Web servers to access information on them. However, when producing Web-based visualizations, you can use these inherent Web structures to augment visualization.

We often use visualization to understand the WWW and improve Web navigation and search tools. Early work in Web visualization represents the beginning of an important new area of research. Gershon et al.[1] developed methods for visualizing hierarchical Web structures and for visually creating a personal information space linking related views of information. Munzner[2] used VRML-based visualization for displaying the WWW's structure and providing links to underlying data. Andrews[3] developed the Harmony system for navigating and visualizating the Hyper-G Internet information system. Mukherjea[4] used Web-based techniques for visualizing the results of a multimedia Web search engine that finds references to both text-based documents and images. Finally, Alper[5] used a Java-based visualization tool for querying and displaying geospatial data within Web environments.

Web-based visualization

While visualizing the Web is a common application of visualization, using the Web to deliver visualization has become a more recent trend. Wood and colleagues[6] used Web-based visualization for providing interactive views

1 Organization Web hierarchy.

of environmental data on the WWW. (They also provided a nice taxonomy and reference model for creating visualization within Web environments.) Walton[7] described and referenced a number of Web-based visualization applications that build upon VRML. Web-based visualization has been adapted for numerous, diverse applications such as chemistry,[8] manufacturing data,[9] and analyzing and debugging parallel computing systems.[10]

Visualization for analysis

Several examples of our work help illustrate the potential benefits and applications of Web-based visualization. Like many institutions, the National Security Agency (NSA) must cope with large amounts of information. We are exploring numerous approaches to understand relationships, trends, and anomalies in large, disparate, multimedia data sources. One of our technology labs—the Community Wide Enterprise Facility (CWEF)—actively explores information visualization techniques to help information analysts understand large, abstract data sets. In addition, like other information-based organizations, NSA is attracted to Web-based technologies. The Web offers a flexible, seamless way to link tools, data, and users. We see our internal Web as a good mechanism for linking such resources.

We view visualization as an integral component in data analysis. For very large data sets, visualization may be the only possible approach. Our development efforts help information analysts find important relationships, anomalies, and trends. Visualization techniques will become even more important as the volume of data continues to increase. Visualization may lead directly to analytic discovery or may be used for data reduction. Direct analytic discovery highlights a key knowledge nugget from a large corpus of data. Using visualization as a data reduction tool, analysts can cull away uninteresting portions and then apply more conventional tools to the remaining data. While we do not see visualization as a panacea, we feel that it is an important analytic tool that will become even more crucial in the future.

Visualization examples

In the following sections, we survey a number of important, well-known information visualization techniques and briefly describe our adaptation and implementation of these approaches to Web-based solutions. This is a snapshot of ongoing work, which we continually mold and evaluate to produce useful Web-based applications. To date, evaluating the effectiveness of these Web-based applications has been subjective. We need more formal usability studies to truly measure their effectiveness within Web environments. However, our initial subjective experience has been favorable.

Hierarchical information

We often use visualization to understand hierarchical information structures. Examples of information hierarchies include organization structure, computer file systems, interlinked Web hierarchies, and communication hierarchies. Common approaches for visualizing information hierarchy are cone trees,[11] tree maps,[12] and hyperbolic browsers.[13] Dynamic arrangements of 3D tree hierarchies, cone trees maximize the visibility of entire trees. However, as cone trees grow very large (approximately 1,000 nodes), they lose their effectiveness due to visual clutter. Recently, cone trees were augmented to work with very large hierarchies by adding numerous interaction techniques and more elaborate layout algorithms.[14] Tree maps represent information by dividing 2D rectangular spaces into recursive boxes. They can represent hierarchy in compact 2D regions very well. Hyperbolic browsers use a focus plus context (fisheye) viewing scheme for large hierarchies that are defined in hyperbolic planes and mapped into compact circular regions. (See the article "Extending Distortion Viewing from 2D to 3D Layout" by Carpendale et al. in this issue for an overview of distortion techniques.)

Most organizations have some form of hierarchical structure that groups people into offices or departments (that is, the ubiquitous organizational chart). Since many organizations use the Web to share relevant information, most Web page hierarchies closely reflect an organization's structure. Our first example of Web-based information visualization uses a variation of cone trees to show organization structure.

The organization hierarchy shown in Figure 1 was created from an organizational listing of NSA's internal Web pages. The nodes are arranged in a cone tree, with the organization's home page at the apex and the rest of the nodes arranged hierarchically. We used cubes to represent individual organization Web pages and color coding to represent various properties such as content, function, or organization type. We also embedded the appropriate hyperlinks within the scene itself. A user simply selects an individual organization (cube), clicks on the associated hyperlink, and jumps to the underlying Web page. That is, node selection triggers the loading of the associated Web document within a browser. (Munzner[2] also reported using linked cone tree structures for viewing the WWW). We created this visualization with an application that reads an on-line document listing the organization's structure and automatically generates the hierarchical cone tree model with embedded links.

Often, we need animation to accurately understand complicated cone tree hierarchies and to view hidden subtrees.[11,14] Our current implementation requires a user to interactively manipulate the view for animation. In the future, we would like to add dynamic restructuring, querying, and viewing. We also plan to integrate this visualization capability with search engines to automatically produce dynamic hierarchies of organization and Web structures.

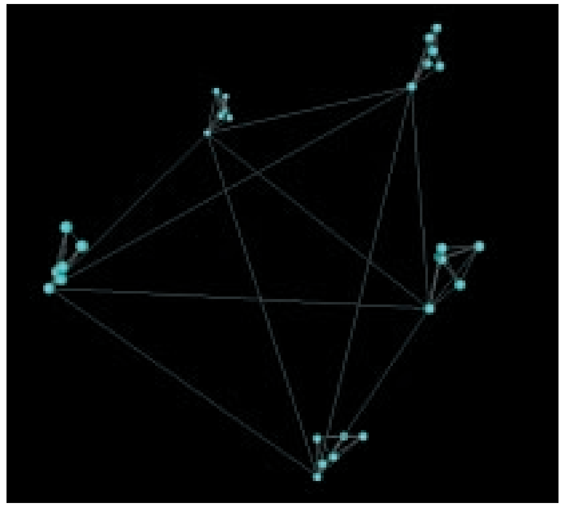
Networks

Frequently, visualization depicts communication network topologies. Such networks may be physical computer networks or logical communication networks. In addition to showing overall structure, dynamic displays can depict changing attributes such as traffic flow, throughput, and bandwidth use. Link-node diagrams are commonly used. In recent years, Eick and colleagues explored numerous approaches to network visualization.[15-17] In addition, a number of commercial products have been developed to visualize and analyze computer networks.

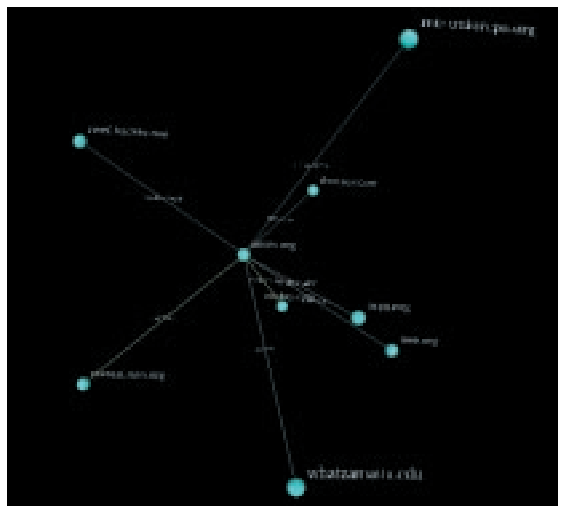
We have also explored network visualization and adapted these techniques for Web-based applications. Figure 2 shows a computer network visualization. The small grouping of nodes indicates logical subnetworks, and each individual node (sphere) represents a local area network (LAN). In this example, we use the level-of-detail feature of VRML. When you move sufficiently close to an individual LAN, it changes representation to a transparent cloud revealing internal structure and displaying individual computers on that LAN.

By using Web-based technologies to visualize computer networks, we can provide links to associated Web-based information. When automatically generating the visualization, we can embed hyperlinks to related node or link information, or encode user actions to trigger CGI-initiated database queries. This allows interaction with the VRML scene to drive queries into an underlying database. The resultant information can be redisplayed as a new visualization, used to update or enhance the existing display, or used to drive a different application. In the future, we plan to use Java directly within the VRML scene to provide a tighter coupling between the visualization and the underlying database. Additional improvements include the use of proximity triggers—actions that are invoked when a user enters, leaves, or moves within a specified region—to help refine data queries and brushing for overlaying textual information. Brushing is an interactive technique that displays associated information when a user moves a cursor over (or brushes) an object. For example, you can use proximity triggers to refine the context of a database query by focusing on a subregion of displayed information as indicated by a user. Brushing provides a high-level interface and feedback for subsequent navigation and data mining operations.

In related work, we have also developed a Web-based network traffic monitoring tool (see Figure 3.) In this example, the user specified a particular computer on a network, shown in the center of the display. From there, all communications to and from that machine are displayed showing links from the specified computer to other computers on the intranet. The network load is color coded on the links between various machines. In this example, we also used multiple levels of detail to represent individual computers (spheres). At higher levels of detail, the spheres become translucent, revealing links and associated information about that machine.



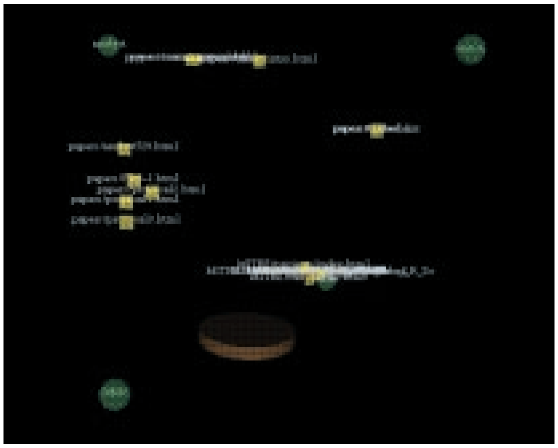
2 **Computer network topology.**



3 **Network traffic monitoring.**

This visualization's true power is that it uses hyperlinks to an underlying database. A user browsing through the network display might select a node to activate a link back to the database. This results in either generating another VRML scene centered on the new node or the display of underlying data within a browser. This visualization application, originally driven from a forms-based Web page, lets users query a database and retrieve the results in a visual form (that is, communications network). The resulting information space can then be further explored through direct manipulation and navigation.

Sometimes very large network topologies produce densely cluttered displays that obscure useful information. This presents a potential problem in network visualization. Techniques for reducing such clutter include



4 **Content-based document clustering and the resulting information space.**

self-organizing schemes and distortion viewing methods. However, applying these techniques to Web-based visualization may not be possible, since they require state knowledge not available in VRML. Our prototype demonstrates some of the features of VRML, which can be used to solve this problem. We use level-of-detail nodes to simplify the display and reduce clutter and complexity. Appropriate detail appears when a viewer approaches a focused region such as a subnetwork. This detail may come from other Web-based information, referenced by or embedded within the VRML model.

Content-based document clustering

Consider the task of sifting through a huge stack of documents and sorting them based on their relevant content. Clearly, this requires an automated approach. The need for such tasks has spawned an active area of research in content-based information retrieval. Such applications produce a large information space that depicts interdocument relationships based on content. You can use visualization to explore and navigate through this resulting information space.

While there have been many developments in this area, the work of Korfhage,[18] Olsen,[19] and Spoerri[20] seems to best capture the ideas of clustering documents into an information space based on their content. Korfhage suggested that an organized display of all documents rather than just the best documents proves important. Olsen (working with Korfhage) developed the Visual Information Browsing Environment (VIBE) system for clustered document arrangement based on key reference points.[19] Spoerri's InfoCrystal also presents an organized arrangement based on content in an elaborate variation of Venn diagrams. Sometimes, presenting abstract information (such as document clusters) in terms of a physical metaphor can be useful. The Themescapes system presents document clusters in the form of geographic terrain or mountains (literally, a landscape of themes).[21]

We have explored several content-based document clustering techniques using our internal Web for sources of documents and for visualization delivery. Figure 4 depicts an information space produced by content-based document clustering using our system. The documents used in this example are the Web pages of our lab. Each Web page—represented by a small yellow tile—includes a link back to the original document. Documents with similar content appear close to each other in this information space. A small, brown disk provides a reference point for orientation within the information space.

Users who wish to view a clustered document space enter up to four keyword query terms that rank the relevance of the information content of each document. The documents are arranged using a simple linear-interpolation of the word count of the documents. We generated a VRML display of the resulting information space—representing the four queries by green spheres at the corners of a regular tetrahedron—and placed documents within the tetrahedron based upon the weights of the query terms in the documents. Even with this simple algorithm, some clustering of information becomes easily visible. Documents grouped close together have relatively similar content based on the query terms. Those documents located near the query spheres have a higher relevance to that particular topic.

Our simple interpolative clustering scheme resembles the VIBE system's[19] scheme. VIBE displays several key terms at vertices on a 2D polygon and locates the relevant documents by interpolating between vertices. By using the third dimension, we can provide additional key terms for greater cluster resolution and pack more information into the document cluster space. The resulting display easily integrates with other VRML visualizations, providing a simple clustering within the context of other visualizations.

Clearly, more sophisticated content clustering methods could be applied. While simple, our method's low complexity could be an advantage for large data sets. Often the purpose of visualization is to provide a quick idea of where to focus and where to quickly locate documents of interest.

Visual Web search

We're also pursuing visually augmenting Web search algorithms. With current search engines, a user enters a keyword and the Web search utility returns an ordered list of Web page references (uniform resource locators or URLs), which are ranked and weighted in a linear list. Instead of displaying a long ordered list of URLs, these Web search results can be displayed in a more meaningful visual form. Typically, you generate the ordered list of search results by consulting a database of Web pages to get a rating of relevance to the query. Unfortunately, this weighting system does not discern the intended context of the user's keyword query. As a result, the query often returns references to information that do not interest the person. Hence, it is often difficult to find relevant information within the context of the user's intentions. In response to these common frustrations, we are investigating using visualization for augmenting Web search engines.

In our system, we use a typical Web search engine but use the ranked results to cluster the resulting search hits (Web documents) into a 3D information space. Using content-based clustering, we generate topic spheres, which group related documents. A modified 2D clustering algorithm (adapted to use spherical geometry) determines the document similarity values by using the most significant key terms from the Web page. The algorithm then clusters the documents based on these similarity scores. The algorithm uses a simplified stress-optimization technique to calculate a vector from the center of the context space for each document and determines the radial distance from the ranking of the original key term of the search. Thus, lower ranked documents are placed farther away from the center of the spherical context space.

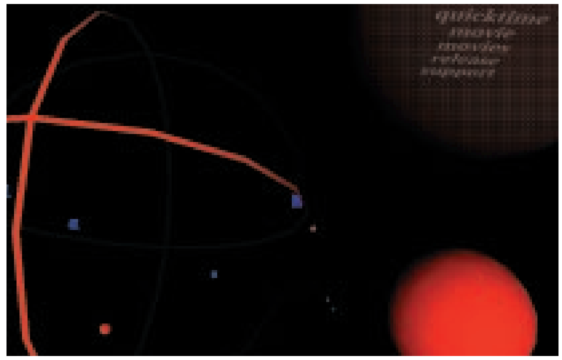
From a distance, solid spheres represent the topic clusters and documents (contained within them) are not visible. This reduces visual clutter and provides convenient landmarks to navigate. Level-of-detail nodes define the topic spheres within VRML. As a user navigates and approaches a sphere, the representation changes to reveal more detail. In this case, the sphere becomes partially transparent, and the terms used to produce the cluster become visible within the sphere. This midlevel representation lets a user easily navigate through the clusters and search for clustered topics of interest. Finally, as the user gets closer to a specific topic cluster, the sphere becomes totally transparent (except for a minimal wireframe of the sphere) and the individual Web documents within the cluster are visible. These specific Web documents are represented by small cubes and have embedded hyperlinks to the actual Web documents. This technique aids in the initial process of scanning a large web of documents for items of interest and to find related items. Our initial prototype of this visual Web search method seems useful. However, we need to conduct quantitative and qualitative usability studies to verify this.

Figure 5 shows a visualization produced by our prototype system. In this example, we performed a visual Web search for the term "animation" against the NSA internal Web pages. The display shows three clusters that result from this query. One of the clusters deals with Quicktime movies and is displayed at the intermediate level with associated query terms. A second cluster displays a set of document links represented by individual cubes. A third is visible as a red sphere in the lower right corner.

Our visual Web search prototype relates to several other systems. LyberWorld supports text retrieval by using relevance spheres to arrange documents in spherical coordinate space.[22] That approach weights vectors with terms of interest. It displays documents within a single sphere based on those topic vectors. Our approach uses a different arrangement technique and adds visible clustering. We also do not allow interactive repositioning based on changing the weights of terms of interest. Mukherjea and colleagues described a system that visualizes the results of a multimedia Web search engine.[4] They also used VRML for Web-based display but only provided scatterplots and tabular views (similar to cityscapes).

Information space metaphors

Information visualization usually deals with abstract data and abstract information spaces. It is often important to place abstract representations within the context



5 **Visual Web search for the term "animation."**

6 **Data vault based on the 3D room information space metaphor.**

of a familiar physical space. Of course, such physical, real-world metaphors should be appropriate for the application and for the individual. Embedding abstract data within a meaningful familiar context can significantly aid in comprehending the data. Finding the appropriate information space metaphor proves a key challenge in producing effective information visualization.

Researchers at Xerox Palo Alto Research Center (PARC) developed several interesting information space metaphors, such as 3D Rooms[23] and WebBook/WebForager.[24] WebBook uses the metaphor of a document library within a 3D work-space and works within Web environments. The Selective Dynamic Manipulation (SDM) system developed by Chuah et al.[25] provides a framework for exploring object-centered metaphors, spatial metaphors, and combinations thereof. Finally, researchers at Pacific Northwest National Laboratory (PNNL) recently described a system for multimedia intelligence analysis based on the 3D room metaphor.[26] This system uses dynamic, interlinked data within the context of a 3D room, which contains numerous multimedia applications.

Using the 3D room metaphor, we developed a Web-based multimedia model of a room, which we refer to as a "data vault" (see Figure 6). Objects in the room link to other Web-based information or are supported by underlying databases. Abstract data can be embedded within the space of the room or displayed on the walls. In this example, we have included the clustered document space from Figure 4 in the upper left-hand corner of the room. A map displayed on the wall references geographic-based data, and a representation of an analyst's desk gives a link to information provided by a topic specialist. Other appropriate visualizations include a hierarchical display for referencing organization structure and a virtual file cabinet to represent a database. We currently provide links to Web-based movies that are associated with data in the room. However, in the future, we plan to include a live video wall by using animated textures and video streams.

Currently a static information space (room), our data vault contains interlinked information. Our future plans to use VRML 2.0 and Java will enable the room to become dynamic. Objects floating in the clustered document space can automatically cross-reference (via brushing) a position on a map or a component of a hierarchy. For example, when you select a document in the cluster space that discusses the activities of a particular organization unit, the unit's location on a map and position within the organization hierarchy may be highlighted with an appropriate label. Alternatively, a browser could display the original underlying data of a document when you select it. While this description of a dynamic data vault remains somewhat speculative, we hope to implement dynamic interactions and real-time data feeds in the future.

Information analysts who do not work in the same physical building could use this virtual multimedia room as a collaborative work space. This approach would let multiple analysts meet within a virtual room full of data. The Web-based approach provides a mechanism for media produced from a variety of sources and users analyzing this information to remain distributed, while providing a central mechanism for interacting with the information.

In addition to 3D rooms, another common physical metaphor that we find useful is geographic terrain. Web-based terrain visualization lets geographic-based applications work in Web environments where they can be interlinked with other information. We have produced numerous VRML-based terrain models, which we use as a base to dynamically overlay and plot analytic data. Using the Web as the delivery mechanism for such applications introduces many opportunities to interlink geographic features with overlaid application data and underlying databases.

Conclusions

In this article we have explored work done by others and by us in Web-based information visualization. We're using the Web as a delivery mechanism for visualization as well as a source of information. While we're encouraged by our experience, we remain cautious. Interlinking visualization components with other Web media and data proves useful. It offers fairly seamless integration of related information for end users. In addition, the widespread use of Web technologies across a range of platforms provides an attractive development path. However, a number of limitations and problems exist.

Current development tools are rather primitive. For example, VRML 1.0 only supports static scenes. Yet, static displays may be sufficient for "broadcasting" important visualization results to a large number of people. While we have achieved a minimal level of dynamic update by using underlying CGI-based programs, such approaches do not provide full interactivity. Integrating Java and VRML 2.0 will improve client-side interaction. Current Web implementations are primarily based on underlying HTTP communication mechanisms, which pose inherent limitations on implementing cooperative, distributed applications. HTTP is a stateless protocol and each interaction must be completely independent of another. Therefore, a server-side application must maintain knowledge of the current state.

One of the current limitations of VRML-based visualization is the variability of the browsers used to display VRML scenes. Since VRML operates across the Web, different browsers on different hardware platforms may be used to view the same data. Unfortunately, not all browsers support all features of VRML. Some browsers may deactivate certain capabilities for performance reasons or let the user control how objects are displayed. Because of this, VRML visualization applications cannot guarantee how a client browser will present the visualization.

Additionally, since VRML is designed to be an efficient language, photorealistic rendering techniques are typically not available within VRML. Advanced lighting and shading models and inter-object reflectance are not standard VRML features. While not a requirement for information visualization, these features can help some users understand intricate relationships within complex data sets. These rendering capabilities will likely improve, but applications that require sophisticated visual cues may not be appropriate for Web-based solutions in the near term.

Performance of Web-based applications remains another important issue. When users query for reference-linked information, the display must promptly provide the results or the user will move on to other information. Since information in Web environments changes so rapidly, applications may not have the luxury of performing computations prior to a query. Complex algorithms may need to take shortcuts, sacrificing accuracy for speed. In addition, critical performance issues associated with slow communications channels and network traffic exist. Slow communication links, such as modem connections, can severely affect the Web's effectiveness. Application developers need to restructure and redesign Web-based applications to accommodate these performance constraints.

The Web is an entirely new medium for visualization and interaction. We're still discovering new Web-based techniques for visualization. We expect the visualization community to continue using Web-based approaches to visualization wherever possible. However, given the inherent limitations, researchers will also need to use traditional approaches for visualization. It's likely that such limitations will disappear in the future. As the Web evolves, we encourage others to explore it for information visualization to further the understanding, potential, and use of this new medium. ■

Acknowledgments

We would like to thank our management and coworkers for providing a supportive environment in which to work and explore. Special thanks to Nahum Gershon for stimulating discussion and presentation suggestions.

References

1. N. Gershon et al., "Visualizing Internet Resources," *Proc. of Information Visualization 95*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1995, pp. 122-128.
2. T. Munzner and P. Burchard, "Visualizing the Structure of the World Wide Web in 3D Hyperbloic Space," Geometry Center, Univ. of Minnesota, on the Web at http://www .geom.umn.edu/docs/research/webviz/webviz/.
3. K. Andrews, "Visualizing Cyberspace: Information Visualization in the Harmony Internet Browser," *Proc. of Information Visualization 95*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1995, pp. 97-104.
4. S. Mukherjea, K. Hirata, and Y. Hara, "Visualizing the Results of Multimedia Web Search Engines," *Proc. of Information Visualization 96*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1996, pp. 64-65.
5. N. Alper and C. Stein, "Geospatial Metadata Querying and Visualization on the WWW Using Java Applets," *Proc. of Information Visualization 96*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1996, pp. 77-84.
6. J.S. Wood, K.W. Brodlie, and H. Wright, "Visualization Over the World Wide Web and its Application to Environmental Data," *Proc. of IEEE Visualization 96*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1996, pp. 81-86.
7. J. Walton, http://www.nag.co.uk/visual/IE/iecbb/VRML2/wm1.html.
8. O. Casher et al., "Advanced VRML-Based Chemistry Applications: 3D Molecular Hyperglossary," *2nd ECC Conf.*, http://www.ch.ic.ac.uk/eccc2/.
9. S. Ressler et al., "Using VRML to Access Manufacturing Data," *Proc .of VRML 97*, ACM Siggraph, ACM Press, New York, 1997, pp. 111-118.
10. N. Osawa, H. Morita, and T. Yuba, "Animation for Performance Debugging of Parallel Computing Systems," *Proc. of VRML 97*, ACM Siggraph, ACM Press, New York, 1997, pp. 103-109.
11. G.G. Robertson, J.D. Mackinlay, and S.K. Card, "Cone Trees: Animated 3D Visualizations of Hierarchical Information," *Proc. of SIGCHI 91*, ACM Press, New York, 1991, pp. 189-194.
12. B. Johnson and B. Shneiderman, "Tree Maps: A Space Filling Approach to the Visualization of Hierarchical Information Structures," *Proc. of IEEE Visualization 91*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1991, pp. 284-291.
13. J. Lamping, R. Rao, and P. Pirolli, "A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies," *Proc. of SIGCHI 95*, ACM Press, New York, 1995, pp. 401-408.
14. J. Carriere and R. Kazman, "Research Report: Interacting with Huge Hierarchies: Beyond Cone Trees," *Proc. of Information Visualization 95*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1995, pp. 204-210.
15. S.G. Eick and G.J. Wilis, "Navigating Large Networks with Hierachies," *Proc. of IEEE Visualization 93*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1993, pp. 204-210.
16. S.G. Eick, "Aspects of Network Visualization," *IEEE Computer Graphics and Applications*, Vol. 16, No. 2, Mar. 1996, pp. 69-72.
17. K. Cox and S.G. Eick, "Case Study: 3D Displays of Internet Traffic," *Proc. of Information Visualization 95*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1995, pp. 129-131.
18. R.R. Korfhage, "To See or Not to See—Is That the Query?," *Proc. of 14th Annual ACM SIGIR Conf.*, ACM Press, New York, 1991, pp. 134-141.
19. K.A. Olsen et al., "Visualization of Document Collection: The VIBE System," *Information Processing Management*, Vol. 29, 1993, pp. 69-81.
20. A. Spoerri, "InfoCrystal: A Visual Tool for Information Retrieval," *Proc. of IEEE Visualization 93*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1993, pp. 150-157.
21. J.A. Wise et al., "Visualizing the Nonvisual: Spatial Analysis and Interaction with Information from Text Documents," *Proc. of Information Visualization 95*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1995, pp. 51-58.
22. M. Hemmje, C. Kunkel, and A. Willet, "LyberWorld—A Visualization User Interface Supporting Full Text Retrieval," *Proc. of 17th Annual ACM SIGIR Conference*, ACM Press, New York, July 1994, pp. 249-259.
23. S. Card, G.G. Robertson, and J.D. Mackinlay, "The Information Visualizer—An Information Workspace," *Proc. of SIGCHI 91*, ACM Press, New York, 1991, pp. 181-188.
24. G.G. Robertson, S. Card, and W. York, "The WebBook and the Web Forager: An Information Workspace for the World Wide Web," *Proc. of SIGCHI 96*, ACM Press, New York, 1996, pp. 11-17.
25. M. Chuah et al., "SDM: Malleable Information Graphics," *Proc. of Information Visualization 95*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1995, pp. 36-42.
26. J.S. Risch et al., "A Virtual Environment for Multimedia Intelligence Data Analysis," *IEEE Computer Graphics and Applications*, Vol. 16, No. 6, Nov. 1996, pp. 33-41.

__Randall M. Rohrer__ is a senior computer scientist at the National Security Agency. His research interests include information visualization, computer graphics, animation, and human-computer interaction. He received a BS in computer science from Pennsylvania State University in 1983 and an MS in computer science from Johns Hopkins University in 1987. He is currently a doctoral student at George Washington University, where he is exploring issues in visualization and shape modeling. He is a member of ACM, Siggraph, and the IEEE Computer Society.

__Edward Swing__ is a senior computer scientist at the National Security Agency. He has developed several applications and prototypes in the fields of geographic systems, computer graphics, and information visualization. He received BS degrees in mathematics and computer science from Rice University in 1986 and an MS from Johns Hopkins University in 1989.

Contact Rohrer at the National Security Agency, E03/CWEF, 9800 Savage Road, Fort Meade, MD 20755-6000, e-mail rohrer@acm.org.