Visualizing the World Wide Web

Alberto O. Mendelzon
University of Toronto
Computer Systems Research Institute
mendelzon@cs.toronto.edu

Abstract

We discuss some principles that we believe are important in creating useful visualizations of the World Wide Web. They are: layout, abstraction, focus, and interaction. We illustrate these points with examples from the work of our group at the University of Toronto.

Introduction

The World Wide Web[BLCL+94] is a large distributed hypermedia network implemented on the global Internet. To give an idea of the size of this network, as of early 1996 one popular index server reports having indexed 21 million Web pages[Cor96].

Not surprisingly, many researchers have proposed visualization as a technology to address the severe problems of disorientation and information overload that are caused by interacting with such a large and chaotic information space. See, for example, [Doe94, And95, AS95, GS95, TL95, HDWB95, MDH95]. In this paper we discuss some principles, based on our experience with information visualization in other domains as well as in the Web domain, that we believe are important in making visualization truly useful. In sum, these are: layout, abstraction, focus, and interaction. We illustrate these points with examples from the work of our group at the University of Toronto.

Why visualize?

Robertson *et al.* [RCR93], in 1993, described information visualization as "[the use of] advancing graphic technology to lower the cost of finding information and accessing it once found." In the context of the Web, that definition is most obviously applicable to the problems encountered by a user trying to find information in the Web. But there are other applications as well; for example, a web site administrator[PB94] may find it useful to visualize relative access frequencies of documents at the site, or most popular paths through a set of documents, or changes in access patterns over time.

Thüring *et al.* [THH95] give a taxonomy of cognitive issues encountered in hypermedia design, where the overall objective is described as "the construction of a mental model that represents the objects and semantic relations described in a text." They list several design principles and relate them to particular cognitive issues. The principle "visualize the structure of the document" is found most relevant to the following cognitive issues: how to provide an adequate overview of the hyperdocument; how to appropriately indicate semantic relations between information units; how to reduce the impression of fragmentation; how to support navigation with respect to direction; and how to support navigation with respect to distance. These ideas systematize and generalize the use of overview diagrams as proposed in many hypermedia systems such as Intermedia[UY89]. While useful for the purpose of hyperdocument *design*, they are perhaps less relevant to the problem of the reader who needs to read a "document" that is not a document, but a collection of many pages written at different times by different people for different purposes. Here fragmentation is inherent, and semantic relations between information units obscure.

In sum, we want to produce Web visualizations because the reasons for doing this in conventional Hypermedia –helping the navigation, bringing out relationships among parts, reducing fragmentation– become even more important in Web documents, although perhaps the objectives must be less ambitious.

In this paper we will concentrate mostly on visualizations that emphasize the graph-like nature of the Web by using some form of node-link diagram. But we should note that there are many other approaches that use radically different visual representations, such as [RCR93, And95, JS91]. Some of these have been applied to Web visualization, some have not.

Visualization principles

Layout

The simplest approach to Web visualization is this: the Web is a graph; there are many algorithms for drawing graphs;

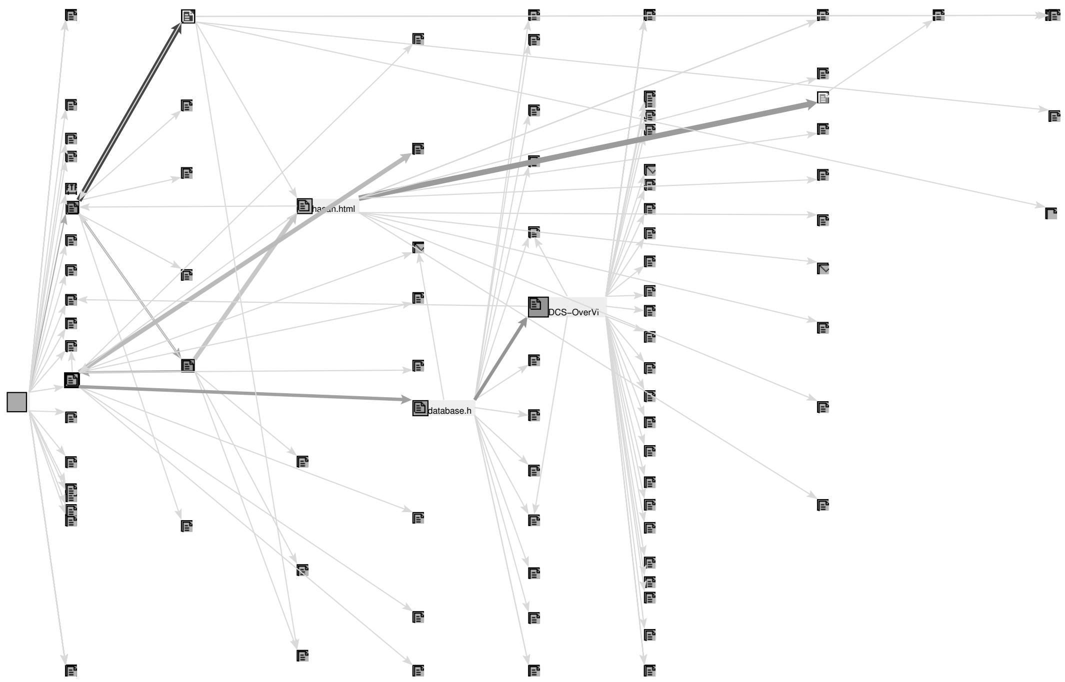

Figure 1: A constrained layout with emphasized nodes.

pick one (or several) and use them to draw some portion of the Web. Problem solved. Of course, this does not work. The first reason is *scale*. While graph drawing is a growing field [DETT94], general techniques that will produce useful visual representations of graphs with as few as several hundred nodes and edges do not exist. The key here is perhaps the word "general;" for well-understood domains and well-specified tasks, research groups such as S. Eick's at AT&T Bell Labs have shown highly effective visualizations of very large graphs [EW93], although these tend to be very expensive computationally.

A second problem is that the layout should depend on the task. For example, when the main purpose of the visualization is to help the user keep track in an organized way of the history of the current browsing session, a hierarchical layout can be useful, and this is what many prototypes in the area provide[GLW+95, MF95, UY89]. But if a web site administrator is trying to analyze a set of documents to understand which ones tend to be browsed together (like Eick [Eic96] does for supermarket purchases), then a spring-based layout[DETT94] can be more appropriate; this is what Tilley and Lamia use in their system[TL95]. And if we are analyzing network traffic to plan communication facilities, we may want a layout where node positions are constrained to be roughly consistent with geography, as described in[GE95].

In Noik's *Graphite* visualizer [Noi96, HGN+95] layout can be constrained by mapping node attributes to horizontal and vertical coordinates. For example, in Figure 1, the horizontal coordinate represents the time when a node was added to the graph, and the vertical layout is computed by a one-dimensional spring algorithm.

Abstraction

Abstraction is an obvious technique for making complex networks manageable by imposing structure on what looks like chaos. Abstraction often goes under the name of *clustering* in the literature. For example, in the Navigational Web Builder[MF95] nodes can be grouped together into higher-level "clusters" based on structural or content-related properties.

In the Hy+ system[CEH+94], abstraction is accomplished by using a form of nested graphs (called *hygraphs* in [Con94]) that can be used to group arbitrary region of the graph to any depth, and can be interactively expanded and contracted. Using the *GraphLog* query language, arbitrary groupings can be defined and displayed at any time. For example, Figure 2 shows part of a Web browsing session where documents are represented by boxes and nodes within the boxes are anchors that point to other documents. This view is not built-in; it was generated by the GraphLog query shown in Figure 3. The left box in Figure 3 changes a `contains` edge from a document to an anchor into a box owned by the document and containing all anchors. The box on the right specifies what components should be displayed: the boxes and two kinds of edges. Details can be found in[HMV95].

Sometimes, abstraction is viewed as a technique to reduce screen clutter and keep the number of objects on the screen small. This may be desirable at times, but in general the need

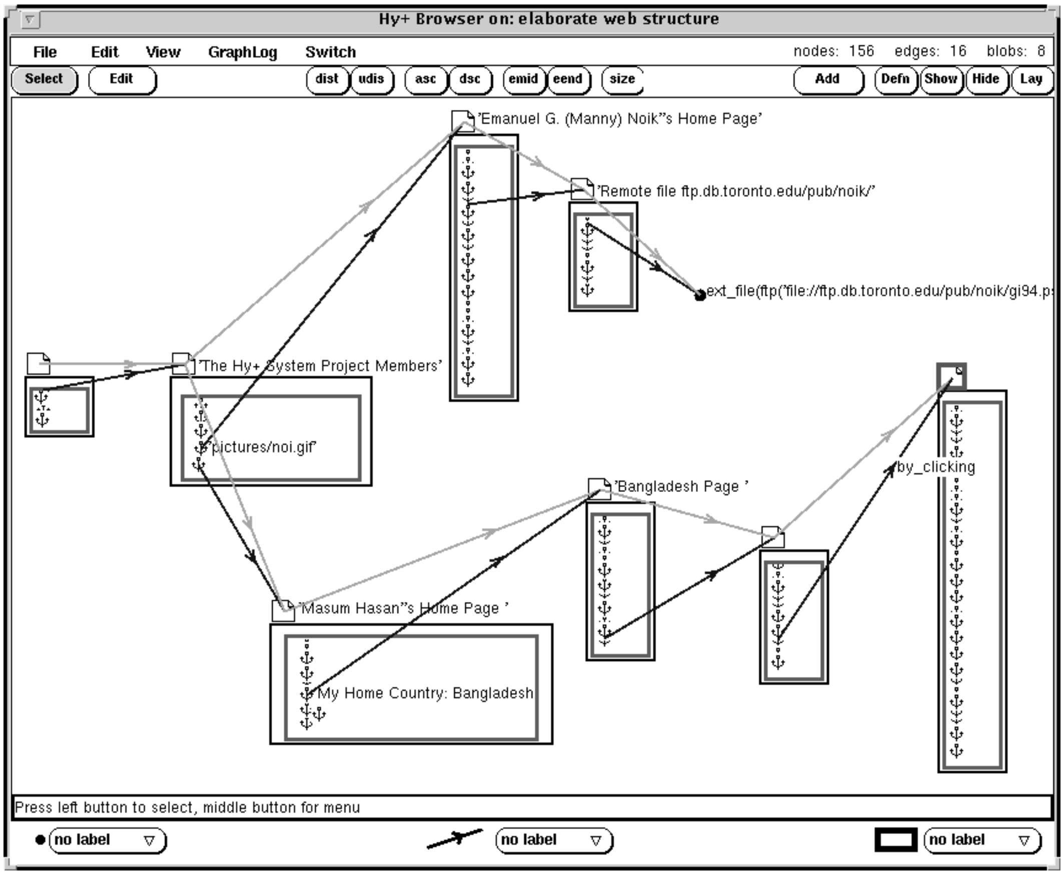

Figure 2: A Hy+ visualization.

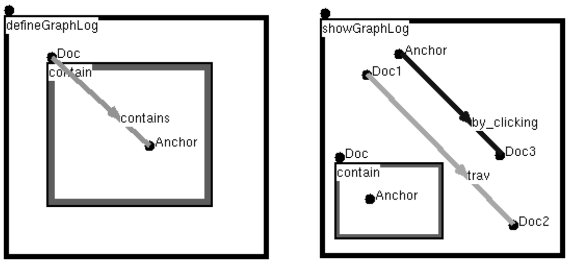

Figure 3: Query that generates the Hy+ view.

to have few things on the screen is a symptom of our inability to design appropriate visualizations that display many objects at the same time. Tufte[Tuf90] is eloquent on this point:

> What about confusing clutter? Information overload? Doesn't data have to be "boiled down" and "simplified"? These common questions miss the point, for the quantity of detail is an issue completely separate from the difficulty of reading. *Clutter and confusion are failures of design, not attributes of information.*

Tufte claims that *layering* and *separation* are among the most powerful devices for enhancing visualizations, visually stratifying various aspects of the data. The use of nested graphs as in Hy+, together with the ability to emphasize and deemphasize dynamically various aspects of a visualization (discussed below under "Interaction") are part of the infrastructure needed to achieve layering in Web visualizations. Bardon[Bar92] proposes a method for managing colours in layered visualizations where objects can dynamically change layers.

Focus

One way of focusing is to select for display only information that is relevant to the task as hand. Providing powerful ways to focus on relevant information was a major thrust of the Hy+ project. The Hy+ approach is to integrate with the visualizer a *query language* in the database tradition, allowing precise declarative specifications of what information is to be visualized and how, using both structural and contents criteria. In particular, Hy+ incorporates a query language called *GraphLog* that is also visual; queries are annotated (hy)graph patterns whose matches on the database can be displayed in different ways. In [CM89], Consens and Mendelzon explored the use of GraphLog queries on hypertext networks, showing that a large number of queries described in the literature could be concisely expressed in GraphLog. More recently, we have used Hy+, as suggested by Figures 2 and 3, to visualize a browsing session and provide various customizable maps and overview diagrams.

Another way of focusing is to emphasize certain parts of the display, while retaining the rest in de-emphasized form to provide context. This is the idea that motivates *fish-eye views* and other emphasis techniques (see [Noi94] for a survey). For example, in Figure 1, nodes that represent "comprehensive" documents are larger; this is accomplished by mapping the link fan-out of a document to the size of the icon that represents it. Other emphasis techniques shown in the picture (but hard to see in this black and white version) include, for example, using different edge colours and widths to represent recency of link traversal.

Interaction

As Huber says[Hub83]: "We see more when we interact with the picture–especially if it reacts instantaneously–than when we merely watch." There are many ways in which direct manipulation can make a Web visualization more informative.

For an example, consider Figure 4 from Noik's *Graphite* system[Noi96]. Much of the explanation has to appeal to the imagination, since the picture is static and interaction is dynamic. First, "brushing" the mouse over a node [BCW87] has the effect of showing the full URL and title for the node (an abbreviated version is displayed otherwise). It also causes certain bars in the histograms at the bottom of the screen to become highlighted. These histograms represent node and link attributes and the frequency distribution of their values in the graph being displayed. For example, for the Web visualization we use node attributes such as size of the document, time when the document was last browsed, time when a link to the document was first discovered, indegree and outdegree; and link attributes such as when the link was first discovered and when it was last traversed. Brushing can also be applied to the histograms themselves, causing the nodes/links corresponding to the bar to be either highlighted, or eliminated from the display. By clicking on histogram bars, the sets of nodes/links corresponding to the bars can be displayed or ellided. Note the similarity to Shneiderman's *dynamic query* paradigm[Shn94]. Node and link attributes can also be dynamically mapped to visual attributes such as horizontal and vertical coordinate, colour hue and saturation, size, thickness, etc.

Another example of interaction is the use of the QRL text search paradigm[GC93] in the Multisurf system[HGN+95]. As shown in Figure 5, the user can click on words or phrases appearing anywhere in a document being browsed, connect these with line segments representing conjunction, and request a search from a set of index servers on the resulting Boolean query. In the figure, the user has requested a search using the Boolean query *"nserc" or ("database" and "visualization") or ("deductive" and "databases").*

Conclusion

Illustrations in the information visualization literature (including the simple figures in this paper) are very far from the visual sophistication and effectiveness of what we find in books like Tufte's. Partly this is due to the limitations of the technology, and partly to the fact that we are trying to *automate* the process of visualization; the more general the domain that we address, the less our pictures will be able to compete with special purpose visualizations carefully crafted by graphic designers. On the other hand, we have a big advantage over pictures in books; our visualizations are dynamic and interactive. We have little doubt that diagrammatic notations and representations of various kinds will play a role of growing importance in the process of accessing large distributed information spaces like the Web, and this paper tries to state some useful design principles for these representations.

Acknowledgements

Manny Noik made many useful comments on an earlier version of this paper. The work described was supported by the Information Technology Research Centre of Ontario and the Natural Sciences and Engineering Research Council of Canada.

References

[And95] Keith Andrews. Visualising cyberspace: Information visualization in the Harmony Internet browser. In *Proceedings '95 Information Visualization*, pages 97–104, Atlanta, October 1995.

[AS95] Eric Z. Ayers and John T. Stasko. Using graphic history in browsing the World Wide Web. In *Proceedings of the Fourth International World Wide Web Conference*, Boston, December 1995. *http://www.w3.org/-pub/Conferences/WWW4/Papers2/270.*

[Bar92] Didier Bardon. Management of color usage in dynamic mapping environments: Balancing semantics, visual ordering and discernability. In *Proceedings of the International Workshop AVI '92*, pages 50–67, Singapore, 1992. World Scientific.

[BCW87] Richard A. Becker, William S. Cleveland, and Allan R. Wilks. Dynamic graphics for data analysis. *Statistical Science*, 2(4):355–395, 1987.

[BLCL+94] Tim Berners-Lee, Robert Cailliau, Ari Luotonen, Henrik Frystyk Nielsen, and Arthur Secret. The World-Wide Web. *Comm. of the ACM*, 37(8):76–82, aug 1994.

[CEH+94] Mariano P. Consens, Frank Ch. Eigler, Masum Z. Hasan, Alberto O. Mendelzon, Emanuel G. Noik, Arthur G. Ryman, and Dimitra Vista. Architecture and applications of the Hy+ visualization system. *IBM Systems Journal*, 33(3):458–476, 1994.

[CM89] Mariano P. Consens and Alberto O. Mendelzon. Expressing structural hypertext queries in graphlog. In

Proc. 2nd. ACM Conference on Hypertext, pages 269–292, Pittsburgh, November 1989.

[Con94] Mariano P. Consens. *Creating and Filtering Structural Data Visualizations Using Hygraph Patterns*. PhD thesis, University of Toronto, Department of Computer Science, June 1994.

[Cor96] Digital Equipment Corporation. Alta Vista: Main page. 1996. *http://altavista.digital.com*.

[DETT94] Giuseppe DiBattista, Peter Eades, Roberto Tamassia, and Ioannis Tollis. Annotated bibliography on graph drawing algorithms. *Computational Geometry: Theory and Applications*, 4:235–282, 1994.

[Doe94] Peter Doemel. WebMap – a graphical hypertext navigation tool. In *Proceedings of the Second International World Wide Web Conference*, Chicago, 10 1994. *http://www.ncsa.uiuc.edu/SDG/IT94/-Proceedings/Searching/doemel/www-fall94.html*.

[Eic96] Stephen G. Eick. Aspects of network visualization. *IEEE Computer Graphics and Applications*, March 1996. In press.

[EW93] Stephen G. Eick and Graham J. Wills. Navigating large networks with hierarchies. In *Proceedings of the Conference on Visualization '93*, pages 204–209, San Jose, October 1993.

[GC93] Gene Golovchinsky and Mark H. Chignell. Queries-R-Links: Graphical markup for text navigation. In *Proceedings InterCHI '93*, Amsterdam, May 1993.

[GE95] Nahum Gershon and Stephen G. Eick. Visualization's new tack: making sense of information. *IEEE Spectrum*, 32(11):38–56, November 1995.

[GLW+95] Naum Gershon, Joshua LeVasseur, Joel Winstead, James Croall, Ari Pernick, and William Ruh. Visualizing internet resources. In *Proceedings '95 Information Visualization*, pages 122–128, Atlanta, October 1995.

[GS95] Brain R. Gaines and Mildred R. Shaw. WebMap: Concept mapping on the Web. In *Proceedings of the Fourth International World Wide Web Conference*, Boston, December 1995. *http://www.w3.org/pub/-Conferences/WWW4/Papers/134*.

[HDWB95] R.J. Hendley, N.S. Drew, A.M. Wood, and R. Beale. Narcissus: Visualizing information. In *Proceedings '95 Information Visualization*, pages 90–96, Atlanta, October 1995.

[HGN+95] Masum Z. Hasan, Gene Golovchinsky, Emanuel G. Noik, Nipon Charoenkitkarn, Mark Chignell, Alberto O. Mendelzon, and David Modjeska. Visual Web surfing with Hy+. In *Proceedings CASCON '95*, Toronto, November 1995. IBM Canada. *ftp://-db.toronto.edu/pub/paperscascon95-multisurf.ps.Z.*

[HMV95] Masum Z. Hasan, Alberto O. Mendelzon, and Dimitra Vista. Visual Web surfing with Hy+. In *Proceedings CASCON '95*, pages 218–227, Toronto, November 1995. IBM Canada. *ftp://db.toronto.edu/pub/papers-cascon95-web-hyplus.ps.Z.*

[Hub83] Peter J. Huber. Statistical graphics: History and overview. In *Proceedings of the Fourth Annual Conference and Exposition of the National Computer Graphics Association*, pages 667–676, 1983.

[JS91] Brian Johnson and Ben Shneiderman. Tree-Maps: A space-filling approach to the visualization of hierarchical information structures. In *Proceedings of IEEE Visualization '91*, pages 284–291, San Diego, October 1991.

[MDH95] Richard Mitchell, David Day, and Lynette Hirschman. Fishing for information on the Internet. In *Proceedings '95 Information Visualization*, pages 105–111, Atlanta, October 1995.

[MF95] S. Mukherjea and J.D. Foley. Visualizing the World Wide Web with the Navigational View Builder. *Computer Networks and ISDN Systems*, 27:1075–1087, 1995.

[Noi94] Emanuel G. Noik. A space of presentation emphasis techniques for visualizing graphs. In *GI '94: Graphics Interface 1994*, pages 225–234, May 1994.

[Noi96] Emanuel G. Noik. *Dynamic Fisheye Views: Combining Dynamic Queries and Mapping with Database Views*. PhD thesis, University of Toronto, Department of Computer Science, 1996. Forthcoming.

[PB94] J.E. Pitkow and K.A. Bharat. WebViz: A tools for World Wide Web access log analysis. In *Proceedings of the First International World Wide Web Conference*, Geneva, Switzerland, May 1994. *http://-www1.cern.ch/PapersWWW94/pitkow-webvis.ps*.

[RCR93] G.G. Robertson, S.K. Card, and C.S. Roberts. Information visualization using 3D interactive animation. *Comm. of the ACM*, 36(4):57–71, apr 1993.

[Shn94] Ben Shneiderman. Dynamic queries for visual information seeking. *IEEE Software*, 11(6):70–77, November 1994.

[THH95] Manfred Thüring, Jörg Hannemann, and Jörg M. Haake. Hypermedia and cognition: Designing for comprehension. *Comm. of the ACM*, 38(8):57–73, aug 1995.

[TL95] Scott R. Tilley and Walter M. Lamia. Personalized information structures II: Hyperstructure hotlists. In *Proceedings of the 13th Annual International Conference on Systems Documentation (SIGDOC '95)*, pages 171–180, Savannah, Georgia, October 1995.

[Tuf90] Edward R. Tufte. *Envisioning Information*. Graphics Press, Cheshire, Connecticut, 1990.

[UY89] K. Utting and N. Yankelovich. Context and orientation in hypermedia networks. *ACM Transactions on Office Information Systems*, 7(1):58–84, 1989.

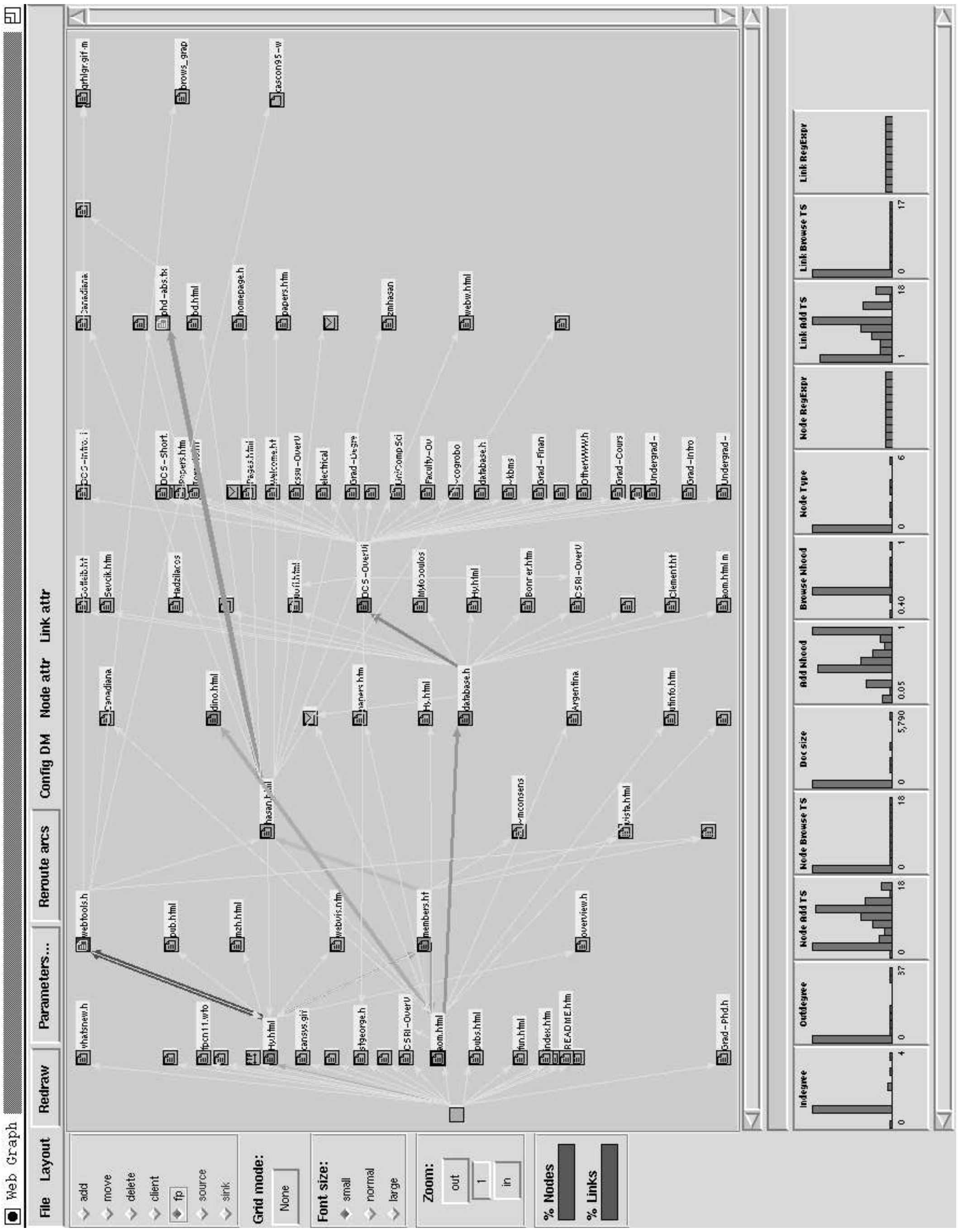

Figure 4: Interacting with a visualization.

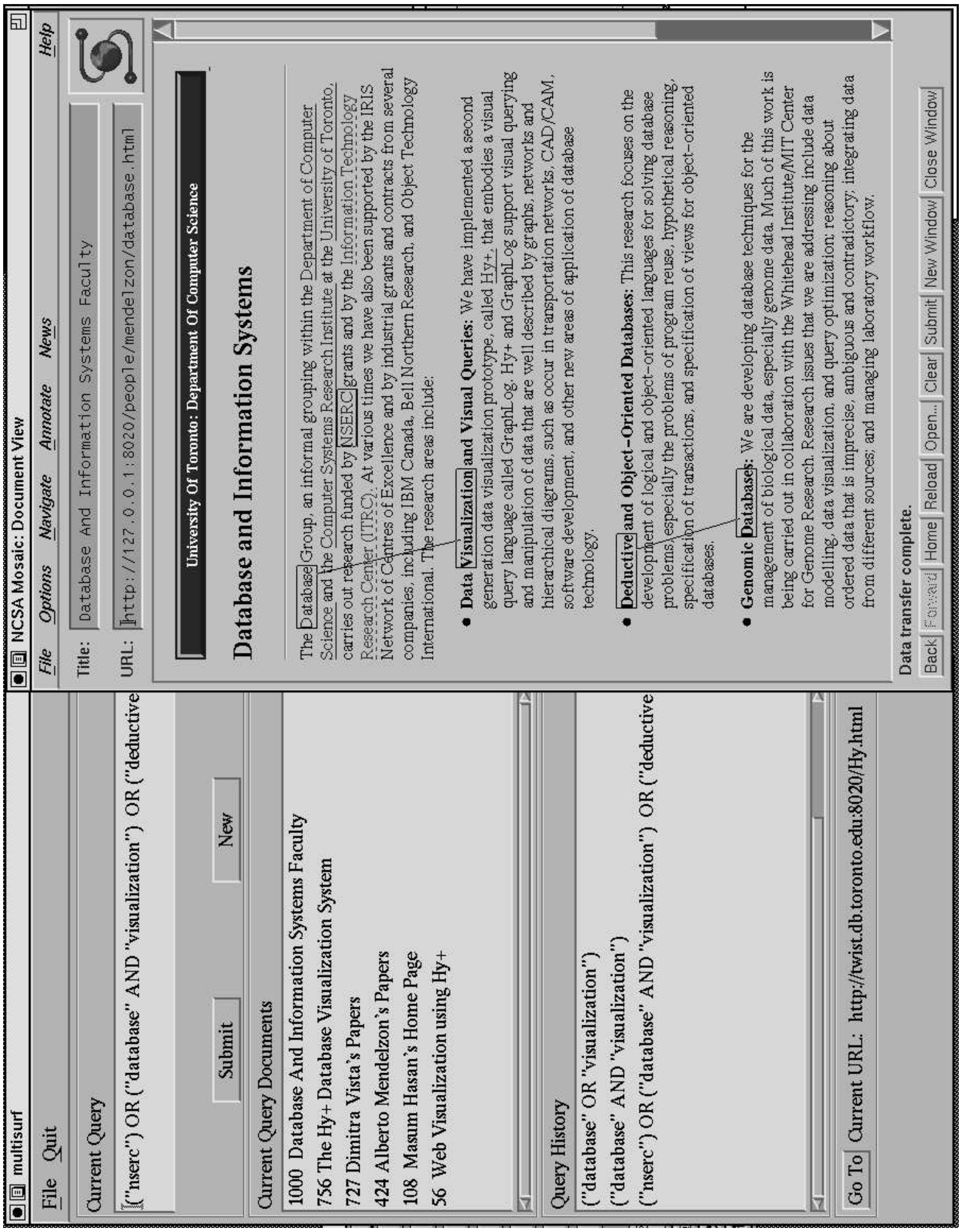

Figure 5: Specifying a Boolean query by graphical markup.