The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations

Ben Shneiderman
Department of Computer Science,
Human-Computer Interaction Laboratory, and Institute for Systems Research
University of Maryland
College Park, Maryland 20742 USA
ben@cs.umd.edu

Abstract

A useful starting point for designing advanced graphical user interfaces is the Visual Information-Seeking Mantra: overview first, zoom and filter, then details on demand. But this is only a starting point in trying to understand the rich and varied set of information visualizations that have been proposed in recent years. This paper offers a task by data type taxonomy with seven data types (one-, two-, three-dimensional data, temporal and multi-dimensional data, and tree and network data) and seven tasks (overview, zoom, filter, details-on-demand, relate, history, and extracts).

> Everything points to the conclusion that the phrase 'the language of art' is more than a loose metaphor, that even to describe the visible world in images we need a developed system of schemata.

E. H. Gombrich *Art and Illusion*, 1959 (p. 76)

1. Introduction

Information exploration should be a joyous experience, but many commentators talk of information overload and anxiety (Wurman, 1989). However, there is promising evidence that the next generation of digital libraries for structured databases, textual documents, and multimedia will enable convenient exploration of growing information spaces by a wider range of users. Visual language researchers and user-interface designers are inventing powerful information visualization methods, while offering smoother integration of technology with task.

The terminology swirl in this domain is especially colorful. The older terms of information retrieval (often applied to bibliographic and textual document systems) and database management (often applied to more structured relational database systems with orderly attributes and sort keys), are being pushed aside by newer notions of information gathering, seeking, or visualization and data mining, warehousing, or filtering. While distinctions are subtle, the common goals reach from finding a narrow set of items in a large collection that satisfy a well-understood information need (known-item search) to developing an understanding of unexpected patterns within the collection (browse) (Marchionini, 1995).

Exploring information collections becomes increasingly difficult as the volume grows. A page of information is easy to explore, but when the information becomes the size of a book, or library, or even larger, it may be difficult to locate known items or to browse to gain an overview.

Designers are just discovering how to use the rapid and high resolution color displays to present large amounts of information in orderly and user-controlled ways. Perceptual psychologists, statisticians, and graphic designers (Bertin, 1983; Cleveland, 1993; Tufte, 1983, 1990) offer valuable guidance about presenting static information, but the opportunity for dynamic displays takes user interface designers well beyond current wisdom.

2. Visual Information Seeking Mantra

The success of direct-manipulation interfaces is indicative of the power of using computers in a more visual or graphic manner. A picture is often cited to be worth a thousand words and, for some (but not all) tasks, it is clear that a visual presentation—such as a map or photograph—is dramatically easier to use than is a textual description or a spoken report. As computer speed and display resolution increase, information visualization and graphical interfaces are likely to have an expanding role. If a map of the United States is displayed, then it should be possible to point rapidly at one of 1000 cities to get tourist information. Of course, a foreigner who knows a city's name (for example, New Orleans), but not its location, may do better with a scrolling alphabetical list.

Visual displays become even more attractive to provide orientation or context, to enable selection of regions, and to provide dynamic feedback for identifying changes (for example, a weather map). Scientific visualization has the power to make atomic, cosmic, and common three-dimensional phenomena (such as heat conduction in engines, airflow over wings, or ozone holes) visible and comprehensible. Abstract information visualization has the power to reveal patterns, clusters, gaps, or outliers in statistical data, stock-market trades, computer directories, or document collections.

Overall, the bandwidth of information presentation is potentially higher in the visual domain than for media reaching any of the other senses. Humans have remarkable perceptual abilities that are greatly under-utilized in current designs. Users can scan, recognize, and recall images rapidly, and can detect changes in size, color, shape, movement, or texture. They can point to a single pixel, even in a megapixel display, and can drag one object to another to perform an action. User interfaces have been largely text-oriented, so as visual approaches are explored, appealing new opportunities are emerging.

There are many visual design guidelines but the basic principle might be summarized as the Visual Information Seeking Mantra:

Overview first, zoom and filter, then details-on-demand
Overview first, zoom and filter, then details-on-demand
Overview first, zoom and filter, then details-on-demand
Overview first, zoom and filter, then details-on-demand
Overview first, zoom and filter, then details-on-demand
Overview first, zoom and filter, then details-on-demand
Overview first, zoom and filter, then details-on-demand
Overview first, zoom and filter, then details-on-demand
Overview first, zoom and filter, then details-on-demand
Overview first, zoom and filter, then details-on-demand

Each line represents one project in which I found myself rediscovering this principle and therefore wrote it down it as a reminder. It proved to be only a starting point in trying to characterize the multiple information-visualization innovations occurring at university, government, and industry research labs.

3. Task by Data Type Taxonomy

To sort out the prototypes and guide researchers to new opportunities, I propose a type by task taxonomy (TTT) of information visualizations. I assume that users are viewing collections of items, where items have multiple attributes. In all seven data types (1-, 2-, 3-dimensional data, temporal and multi-dimensional data, and tree and network data) the items have attributes and a basic search task is to select all items that satisfy values of a set of attributes. An example task would be finding all divisions in an organization structure that have a budget greater than $500,000.

The data types are on the left side of the TTT characterize the task-domain information objects and are organized by the problems users are trying to solve. For example, in two-dimensional information such as maps, users are trying to grasp adjacency or navigate paths, whereas in tree-structured information users are trying to understand parent/child/sibling relationships. The tasks across the top of the TTT are task-domain information actions that users wish to perform.

The seven tasks are at a high level of abstraction. More tasks and refinements of these tasks would be natural next steps in expanding this table. The seven tasks are:

Overview: Gain an overview of the entire collection.
Zoom : Zoom in on items of interest
Filter: filter out uninteresting items.
Details-on-demand: Select an item or group and get details when needed.
Relate: View relationships among items.
History: Keep a history of actions to support undo, replay, and progressive refinement.
Extract: Allow extraction of sub-collections and of the query parameters.

Further discussion of the tasks follows the descriptions of the seven data types:

1-dimensional: linear data types include textual documents, program source code, and alphabetical lists of names which are all organized in a sequential manner. Each item in the collection is a line of text containing a string of characters. Additional line attributes might be the date of last update or author name. Interface design issues include what fonts, color, size to use and what overview, scrolling, or selection methods can be used. User problems might be to find the number of items, see items having certain attributes (show only lines of a document that are section titles, lines of a program that were changed from the previous version, or people in a list who are older than 21 years), or see an item with all its attributes.

Examples: An early approach to dealing with large 1-dimensional data sets was the bifocal display which provided detailed information in the focus area and less information in the surrounding context area (Spence and Apperley, 1982). In their example, the selected issue of a scientific journal had details about each article, the older and newer issues of the journal were to the left and right on the bookshelf with decreasing space. Another effort to visualize 1-dimensional data showed the attribute values of each thousands of item in a fixed-sized space using a scrollbar-like display called value bars (Chimera, 1992). Even greater compressions were accomplished in compact displays of tens of thousands of lines of program source code (SeeSoft, Eick et al., 1992) or textual documents (Document Lens, Robertson and Mackinlay, 1993; Information mural, Jerding and Stasko, 1995).

2-dimensional: planar or map data include geographic maps, floorplans, or newspaper layouts. Each item in the collection covers some part of the total area and may be rectangular or not. Each item has task-domain attributes such as name, owner, value, etc. and interface-domain features such as size, color, opacity, etc. While many systems adopt a multiple layer approach to dealing with map data, each layer is 2-dimensional. User problems are to find adjacent items, containment of one item by another, paths between items, and the basic tasks of counting, filtering, and details-on-demand.

Examples: Geographic Information Systems are a large research and commercial domain (Laurini and Thompson, 1992; Egenhofer and Richards, 1993) with numerous systems available. Information visualization researchers have used spatial displays of document collections (Korfhage, 1991; Hemmje et al., 1993; Wise et al., 1995) organized proximally by term co-occurrences.

3-dimensional: real-world objects such as molecules, the human body, and buildings have items with volume and some potentially complex relationship with other items. Computer-assisted design systems for architects, solid modelers, and mechanical engineers are built to handle complex 3-dimensional relationships. Users' tasks deal with adjacency plus above/below and inside/outside relationships, as well as the basic tasks. In 3-dimensional applications users must cope with understanding their position and orientation when viewing the objects, plus the serious problems of occlusion. Solutions to some of these problems are proposed in many prototypes with techniques such as overviews, landmarks, perspective, stereo display, transparency, and color coding.

Examples: Three-dimensional computer graphics and computer-assisted design are large topics, but information visualization efforts in three dimensions are still novel. Navigating high resolution images of the human body is the challenge in the National Library of Medicine's Visible Human project (North et al., 1996). Some applications have attempted to present 3-dimensional versions of trees (Robertson et al., 1993), networks (Fairchild et al., 1988), or elaborate desktops (Card et al., 1996).

Temporal: time lines are widely used and vital enough for medical records, project management, or historical presentations to create a data type that is separate from 1-dimensional data. The distinction in temporal data is that items have a start and finish time and that items may overlap. Frequent tasks include finding all events before, after, or during some time period or moment, plus the basic tasks.

Examples: Many project management tools exist, but novel visualizations of time include the perspective wall (Robertson et al., 1993) and LifeLines (Plaisant et al., 1996). LifeLines shows a youth history keyed to the needs of the Maryland Department of Juvenile Justice, but is intended to present medical patient histories as a compact overview with selectable items to get details-on-demand. Temporal data visualizations appear in systems for editing video data or composing animations such as Macromedia Director.

Multi-dimensional: most relational and statistical databases are conveniently manipulated as multi-dimensional data in which items with n attributes become points in a n-dimensional space. The interface representation can be 2-dimensional scattergrams with each additional dimension controlled by a slider (Ahlberg and Shneiderman, 1994). Buttons can used for attribute values when the cardinality is small, say less than ten. Tasks include finding patterns, clusters, correlations among pairs of variables, gaps, and outliers. Multi-dimensional data can be represented by a 3-dimensional scattergram but disorientation (especially if the users point of view is inside the cluster of points) and occlusion (especially if close points are represented as being larger) can be problems. The technique of parallel coordinates is a clever innovation which makes some tasks easier, but takes practice for users to comprehend (Inselberg, 1985).

Examples: The early HomeFinder developed dynamic queries and sliders for user-controlled visualization of multi-dimensional data (Williamson and Shneiderman, 1992). The successor FilmFinder refined the techniques (Ahlberg and Shneiderman, 1994) for starfield displays (zoomable, color coded, user-controlled scattergrams), and laid the basis for the commercial product Spotfire (Ahlberg and Wistrand, 1995). Extrapolations include the Aggregate Manipulator (Goldstein and Roth, 1994), movable filters (Fishkin and Stone, 1995), and Selective Dynamic Manipulation (Chuah et al., 1995). Related works include VisDB for multidimensional database visualization (Keim and Kreigal, 1994), the spreadsheet-like Table Lens (Rao and Card, 1994) and the multiple linked histograms in the Influence Explorer (Tweedie et al., 1996).

Tree: hierarchies or tree structures are collections of items with each item having a link to one parent item (except the root). Items and the links between parent and child can have multiple attributes. The basic tasks can be applied to items and links, and tasks related to structural properties become interesting, for example, how many levels in the tree? or how many children does an item have? While it is possible to have similar items at leaves and internal nodes, it is also common to find different items at each level in a tree. Fixed level trees with all leaves equidistant from the root and fixed fanout trees with the same number of children for every parent are easier to deal with. High fanout (broad) and small fanout (deep) trees are important special cases. Interface representations of trees can use an outline style of indented labels used in tables of contents (Chimera and Shneiderman, 1993), a node and link diagram, or a treemap, in which child items are rectangles nested inside parent rectangles.

Examples: Tree-structured data has long been displayed with indented outlines (Egan et al., 1989) or with connecting lines as in many computer-directory file managers. Attempts to show large tree structures as node and link diagrams in compact forms include the 3-dimensional cone and cam trees (Robertson et al., 1993; Carriere and Kazman, 1995), dynamic pruning in the TreeBrowser (Kumar et al., 1995), and the appealingly animated hyperbolic trees (Lamping et al., 1995). A novel space-filling mosaic approach shows an arbitrary sized tree in a fixed rectangular space (Shneiderman, 1992; Johnson and Shneiderman, 1991). The treemap approach was successfully applied to computer directories, sales data, business decision-making (Asahi et al., 1995), and web browsing (Mitchell et al., 1995; Mukherjea et al., 1995), but users take 10-20 minutes to accommodate to complex treemaps.

Network: sometimes relationships among items cannot be conveniently captured with a tree structure and it is useful to have items linked to an arbitrary number of other items. While many special cases of networks exist (acyclic, lattices, rooted vs. un-rooted, directed vs. undirected) it seems convenient to consider them all as one data type. In addition to the basic tasks applied to items and links, network users often want to know about shortest or least costly paths connecting two items or traversing the entire network. Interface representations include a node and link diagram, and a square matrix of the items with the value of a link attribute in the row and column representing a link.

Examples: Network visualization is an old but still imperfect art because of the complexity of relationships and user tasks. Commercial packages can handle small networks or simple strategies such as Netmap's layout of nodes on a circle with links criss-crossing the central area. An ambitious 3-dimensional approach was an impressive early accomplishment (Fairchild et al., 1988), and new interest in this topic has been spawned by attempts to visualize the World Wide Web (Andrews, 1995; Hendley et al., 1995).

These seven data types reflect are an abstraction of the reality. There are many variations on these themes (2 1/2 or 4-dimensional data, multitrees,...) and many prototypes use combinations of these data types. This taxonomy is useful only if it facilitates discussion and leads to useful discoveries. Some idea of missed opportunities emerges in looking at the tasks and data types in depth:

Overview: Gain an overview of the entire collection. Overview strategies include zoomed out views of each data type to see the entire collection plus an adjoining detail view. The overview contains a movable field-of-view box to control the contents of the detail view, allowing zoom factors of 3 to 30. Replication of this strategy with intermediate views enables users to reach larger zoom factors. Another popular approach is the fisheye strategy (Furnas, 1986) which has been applied most commonly for network browsing (Sarkar and Brown, 1994; Bartram et al., 1995). The fisheye distortion magnifies one or more areas of the display, but zoom factors in prototypes are limited to about 5. Although query language facilities made it difficult to gain an overview of a collection, information visualization interfaces support some overview strategy, or should. Adequate overview strategies are a useful criteria to look for. Along with an overview plus detail (also called context plus focus) view there is a need for navigation tools to pan or scroll through the collection.

Zoom: Zoom in on items of interest. Users typically have an interest in some portion of a collection, and they need tools to enable them to control the zoom focus and the zoom factor. Smooth zooming helps users preserve their sense of position and context. Zooming could be on one dimension at a time by moving the zoombar controls or by adjusting the size of the field-of -view box. A very satisfying way to zoom in is by pointing to a location and issuing a zooming command, usually by clicking on a mouse button for as long as the user wishes (Bederson and Hollan, 1993). Zooming in one dimension has proven useful in starfield displays (Jog and Shneiderman, 1995).

Filter: filter out uninteresting items. Dynamic queries applied to the items in the collection is one of the key ideas in information visualization (Ahlberg et al., 1992; Williamson and Shneiderman, 1992). By allowing users to control the contents of the display, users can quickly focus on their interests by eliminating unwanted items. Sliders, buttons, or other control widgets coupled to rapid display update (less than 100 milliseconds) is the goal, even when there are tens of thousands of displayed items.

Details-on-demand: Select an item or group and get details when needed. Once a collection has been trimmed to a few dozen items it should be easy to browse the details about the group or individual items. The usual approach is to simply click on an item to get a pop-up window with values of each of the attributes. In Spotfire, the details-on-demand window can contain HTML text with links to further information.

Relate: View relationships among items. In the FilmFinder (Ahlberg and Shneiderman, 1994) users could select an attribute, such as the film's director, in the details-on-demand window and cause the director alphaslider to be reset to the director's name, thereby displaying only films by that director. Similarly, in SDM (Chuah et al., 1995), users can select an item and then highlight items with similar attributes or in LifeLines (Plaisant et al., 1996) users can click on a medication and see the related visit report, prescription, and lab test. Designing user interface actions to specify which relationship is to be manifested is still a challenge. The Influence Explorer (Tweedie et al., 1996) emphasizes exploration of relationships among attributes. and the Table Lens emphasizes finding correlations among pairs of numerical attributes (Rao and Card, 1994).

History : Keep a history of actions to support undo, replay, and progressive refinement. It is rare that a single user action produces the desired outcome. Information exploration is inherently a process with many steps, so keeping the history of actions and allowing users to retrace their steps is important. However, most prototypes fail to deal with this requirement. Maybe they are reflecting the current state of graphic user interfaces, but designers would be better to follow information retrieval systems which typically preserve the sequence of searches so that they can be combined or refined.

Extract: Allow extraction of sub-collections and of the query parameters. Once users have obtained the item or set of items they desire, it would be useful to be able to extract that set and save it to a file in a format that would facilitate other uses such as sending by email, printing, graphing, or insertion into a statistical or presentation package. An alternative to saving the set, they might want to save, send, or print the settings for the control widgets. Very few prototypes support this action, although Roth's recent work on Visage provides an elegant capability to extract sets of items and simply drag-and-drop them into the next application window.

The attraction of visual displays, when compared to textual displays, is that they make use of the remarkable human perceptual ability for visual information. Within visual displays, there are opportunities for showing relationships by proximity, by containment, by connected lines, or by color coding. Highlighting techniques (for example, bold-face text or brightening, inverse video, blinking, underscoring, or boxing) can be used to draw attention to certain items in a field of thousands of items. Pointing to a visual display can allow rapid selection, and feedback is apparent. The eye, the hand, and the mind seem to work smoothly and rapidly as users perform actions on visual displays.

4. Advanced Filtering

Users have highly varied needs for filtering features. The dynamic queries approach of adjusting numeric range sliders, alphasliders for names or categories, or buttons for small sets of categories is appealing to many users for many tasks (Shneiderman, 1994). Dynamic queries might be called *direct-manipulation queries*, since they share the same concepts of visual display of actions (the sliders or buttons) and objects (the query results in the task-domain display); the use of rapid, incremental, and reversible actions; and the immediate display of feedback (less than 100 msec). Additional benefits are no error messages and the encouragement of exploration.

Dynamic queries can reveal global properties as well as assist users in answering specific questions. As the database grows, it is more difficult to update the display fast enough, and specialized data structures or parallel computation are required.

The dynamic-query approach to the chemical table of elements was tested in an empirical comparison with a form-fill-in query interface. The counterbalanced-ordering within-subjects design with 18 chemistry students showed strong advantages for the dynamic queries in terms of faster performance and lower error rates (Ahlberg et al., 1991).

Dynamic queries usually permit OR combinations within an attribute with AND combination of attributes across attributes (conjunct of disjuncts). This is adequate for many situations since rapid multiple sequential queries allow users to satisfy their information needs. Commercial information-retrieval systems, such as DIALOG or Lexis/Nexis, permit complex *Boolean expressions* with parentheses, but widespread adoption has been inhibited by the difficulty of using them. Numerous proposals have been put forward to reduce the burden of specifying complex Boolean expressions (Reisner, 1988). Part of the confusion stems from informal English usage where a query such as List all employees who live in New York and Boston would result in an empty list because the "and" would be interpreted as an intersection; only employees who live in *both* cities would qualify! In English, "and" usually expands the options; in Boolean expressions, AND is used to narrow a set to the intersection of two others. Similarly, in the English "I'd like Russian or Italian salad dressing," the "or" is exclusive, indicating that you want one or the other but not both; in Boolean expressions, an OR is inclusive, and is used to expand a set.

The desire for *full Boolean expressions*, including nested parentheses and NOT operators, led us toward novel metaphors for query specification. *Venn diagrams* (Michard, 1982), *decision tables* (Greene et al., 1990), and the innovative InfoCrystal (Spoerri, 1993) have been used, but these both become confusing as query complexity increases. We sought to support arbitrarily complex Boolean expressions with a graphical specification. Our approach was to apply the metaphor of water flowing from left to right through a series of pipes and filters, where each filter lets through only the appropriate documents, and the pipe layout indicates relationships of AND or OR. (Young and Shneiderman, 1993)

In this filter-flow model, ANDs are shown as a linear sequence of filters, suggesting the successive application of required criteria. As the flow passes through each filter, it is reduced, and the visual feedback shows a narrower bluish stream of water. ORs are shown two ways: within an attribute, multiple values can be selected in a single filter; and across multiple attributes, filters are arranged in parallel paths. When the parallel paths converge, the width of the flow reflects the size of the union of the document sets.

Negation was handled by a NOT operator that, when selected, inverts all currently selected items in a filter. For example, if California and Georgia were selected and then the NOT operator was chosen, those two states would become deselected and all the other states would become selected. Finally, clusters of filters and pipes can be made into a single labeled filter. This facility ensures that the full query can be shown on the display at once, and allows clusters to be saved in a library for later reuse.

We believe that this approach can help novices and intermittent users to specify complex Boolean expressions and to learn Boolean concepts. A usability study was conducted with 20 subjects with little experience using Boolean algebra. The prototype filter-flow interface showed statistically significant improved performance against a textual interface for comprehension and composition tasks. The filter-flow interface was preferred by all 20 subjects.

5. Summary

Novel graphical and direct-manipulation approaches to query formulation and information visualization are now possible. While research prototypes have typically dealt with only one data type (1-, 2-, 3-dimensional data, temporal and multi-dimensional data, and tree and network data), successful commercial products will have to accommodate several. These products will need to provide smooth integration with existing software and support the full task list: Overview, zoom, filter, details-on-demand, relate, history, and extract. These ideas are attractive because they present information rapidly and allow for rapid user-controlled exploration. If they are to be fully effective, some of these approaches require novel data structures, high-resolution color displays, fast data retrieval, specialized data structures, parallel computation, and some user training.

Although the computer contributes to the information explosion, it is potentially the magic lens for finding, sorting, filtering, and presenting the relevant items. Search in complex structured documents, graphics, images, sound, or video presents grand opportunities for the design of user interfaces and search engines to find the needle in the haystack. The novel-information exploration tools—such as dynamic queries, treemaps, fisheye views, parallel coordinates, starfields, and perspective walls—are but a few of the inventions that will have to be tamed and validated.

References

Ahlberg, Christopher and Shneiderman, Ben, Visual information seeking: Tight coupling of dynamic query filters with starfield displays, *Proc. ACM CHI94 Conference: Human Factors in Computing Systems*, (1994), 313-321 + color plates.

Ahlberg, Christopher and Shneiderman, Ben, AlphaSlider: A compact and rapid selector, *Proc. ACM CHI94 Conference Human Factors in Computing Systems*, (1994), 365-371.

Ahlberg, Christopher, Williamson, Christopher, and Shneiderman, Ben, Dynamic queries for information exploration: An implementation and evaluation, *Proc. ACM CHI'92: Human Factors in Computing Systems*, (1992), 619-626.

Ahlberg, Christopher and Wistrand, Erik, IVEE: An information visualization & exploration environment, *Proc. IEEE Information Visualization '95*, (1995), 66-73.

Andrew, Keith, Visualising cyberspace: Information visualisation in the Harmony internet browser, *Proc. IEEE Information Visualization '95*, (1995), 97-104.

Asahi, T., Turo, D., and Shneiderman, B., Using treemaps to visualize the analytic hierarchy process, *Information Systems Research 6*, 4 (December 1995), 357-375.

Bartram, Lyn, Ho, Albert, Dill, John, and Henigman, Frank, The continuous zoom: A constrained fisheye technique for viewing and navigating large information spaces, *Proc. ACM User Interface Software and Technology '95*, (1995), 207-215.

Becker, Richard A., Eick , Stephen G., and Wilks, Allan R. Visualizing Network Data, *IEEE Transactions on Visualization and Computer Graphics 1*, 1 (March 1995), 16-28.

Bederson, Ben B. and Hollan, James D., PAD++: A zooming graphical user interface for exploring alternate interface physics, *Proc. ACM User Interfaces Software and Technology '94* (1994), 17-27.

Bertin, Jacques, *Semiology of Graphics*, University of Wisconsin Press, Madison, WI (1983)

Card, Stuart K., Robertson, George G., and York, William, The WebBook and the WebForager: An information workspace for the World-Wide Web, *Proc. ACM CHI96 Conference: Human Factors in Computing Systems*, (1996), 111-117.

Carriere, Jeremy and Kazman, Rick, Interacting with huge hierarchies: Beyond cone trees, *Proc. IEEE Information Visualization '95*, (1995), 74-81.

Chimera, Richard, Value bars: An information visualization and navigation tool for multiattribute listings, *Proc. ACM CHI92 Conference: Human Factors in Computing Systems*, (1992), 293-294.

Chimera, Richard and Shneiderman, Ben, Evaluating three user interfaces for browsing tables of contents, *ACM Transactions on Information Systems 12*, 4 (October 1994).

Chuah, Mei C., Roth, Steven F., Mattis, Joe, and Kolojejchcik, John, SDM: Malleable Information Graphics, *Proc. IEEE Information Visualization '95*, (1995), 66-73.

Cleveland, William, *Visualizing Data*, Hobart Press, Summit, NJ (1993).

Egan, Dennis E., Remde, Joel R., Gomez, Louis M., Landauer, Thomas K., Eberhardt, Jennifer, and Lochbum, Carol C., Formative design-evaluation of SuperBook, *ACM Transactions on Information Systems 7*, 1 (January 1989), 30–57.

Egenhofer, Max and Richards, J., Exploratory access to geographic data based on the map-overlay metaphor, *Journal of Visual Languages and Computing* 4, 2 (1993), 105-125.

Eick, Stephen G. , Steffen, Jospeh L., and Sumner, Jr., Eric E., SeeSoft- A tool for visualizing line-oriented software statistics, *IEEE Transactions on Software Engineering 18*, 11 (1992) 957-968.

Eick, Stephen G. and Wills, Graham J., Navigating Large Networks with Hierarchies, *Proc. IEEE Visualization '93 Conference*, (1993), 204--210.

Fairchild, Kim M., Poltrock, Steven E., and Furnas, George W., SemNet: Three-dimensional representations of large knowledge bases, In Guindon, Raymonde (Editor), *Cognitive Science and its Applications for Human-Computer Interaction*, Lawrence Erlbaum, Hillsdale, NJ (1988), 201-233.

Fishkin, Ken and Stone, Maureen C., Enhanced dynamic queries via movable filters, *Proc. ACM CHI95 Conference: Human Factors in Computing Systems*, (1995), 415-420.

Furnas, George W., Generalized fisheye views, *Proc. ACM CHI86 Conference: Human Factors in Computing Systems*, (1986), 16-23.

Goldstein, Jade and Roth, Steven F, Using aggregation and dynamic queries for exploring large data sets, *Proc. ACM CHI95 Conference: Human Factors in Computing Systems*, (1995), 23-29.

Greene, S. L., Devlin, S. J., Cannata, P. E., and Gomez, L. M., No IFs, ANDs, or ORs: A study of database querying, *International Journal of Man–Machine Studies 32* (March 1990), 303–326.

Hendley, R. J., Drew, N. S., Wood, A. S., Narcissus: Visualizing information, *Proc. IEEE Information Visualization '95*, (1995), 90-96.

Humphrey, Susanne M. and Melloni, Biagio John, *Databases: A Primer for Retrieving Information by Computer*, Prentice-Hall, Englewood Cliffs, NJ (1986).

Inselberg, Alfred, The plane with parallel coordinates, *The Visual Computer 1* (1985), 69-91.

Jerding, Dean F. and Stasko, John T., The information mural: A technique for displaying and navigating large information spaces, *Proc. IEEE Information Visualization '95*, 43-50.

Jog, Ninad and Shneiderman, Ben, Information visualization with smooth zooming on an starfield display, *Proc. IFIP Conf. Visual Databases 3*, Chapman and Hall, London (1995), 1-10.

Johnson, Brian, and Shneiderman, Ben, Tree-maps: A space-filling approach to the visualization of hierarchical information structures, *Proc. IEEE Visualization'91*, (1991), 284–291.

Keim, D. A. and Kriegal, H., VisDB: Database exploration using multidimensional visualization, *IEEE Computer Graphics and Applications* (September 1994), 40-49.

Korfhage, Robert, To see or not to see -- Is that the query?, *Communications of the ACM 34* (1991), 134-141.

Lamping, John, Rao, Ramana, and Pirolli, Peter, A focus + context technique based on hyperbolic geometry for visualizing large hierarchies, *Proc. ACM CHI95 Conference: Human Factors in Computing Systems*, (1995), 401-408

Laurini, R. and Thompson, D., *Fundamentals of Spatial Information Systems*, Academic Press, New York, NY (1992).

Marchionini, Gary, *Information Seeking in Electronic Environments*, Cambridge University Press, UK (1995).

Michard, A., A new database query language for non-professional users: Design principles and ergonomic evaluation, *Behavioral and Information Technology 1*, 3 (July–September 1982), 279–288.

Mitchell, Richard, Day, David, and Hirschman, Lynette, Fishing for information on the internet, *Proc. IEEE Information Visualization '95*, (1995), 105-111.

Mukherjea, Sougata, Foley, James D., and Hudson, Scott, Visualizing complex hypermedia networks through multiple hierarchical views, *Proc. ACM CHI95 Conference: Human Factors in Computing Systems*, (1995), 331-337 + color plate.

North, Chris, Shneiderman, Ben, and Plaisant, Catherine, User controlled overviews of an image library: A case study of the Visible Human, *Proc. 1st ACM International Conference on Digital Libraries* (1996), 74-82.

Pirolli, Peter, Schank, Patricia, Hearst, Marti, and Diehl, Christine, Scatter/gather browsing communicates the topic structure of a very large text collection, *Proc. ACM CHI96 Conference*, (1996), 213-220.

Plaisant, Catherine, Rose, Anne, Milash, Brett, Widoff, Seth, and Shneiderman, Ben, LifeLines: Visualizing personal histories, *Proc. ACM CHI96 Conference: Human Factors in Computing Systems*, (1996), 221-227, 518.

Rao, Ramana and Card, Stuart K., The Table Lens: Merging graphical and symbolic representations in an interactive focus + context visualization for tabular information, *Proc. ACM CHI94 Conference: Human Factors in Computing Systems*, (1994), 318-322.

Reisner, Phyllis, Query languages. In Helander, Martin (Editor), *Handbook of Human–Computer Interaction*, North-Holland, Amsterdam, The Netherlands (1988), 257–280.

Robertson, George G., Card, Stuart K., and Mackinlay, Jock D., Information visualization using 3-D interactive animation, *Communications of the ACM 36*, 4 (April 1993), 56-71.

Robertson George G. and Mackinlay, Jock D., The document lens, *Proc. 1993 ACM User Interface Software and Technology*, (1993), 101-108.

Sarkar, Manojit and Brown, Marc H., Graphical fisheye views, *Communications of the ACM 37*, 12 (July 1994), 73–84.

Shneiderman, Ben, Tree visualization with tree-maps: A 2-d space-filling approach, *ACM Transactions on Graphics 11*, 1 (January 1992), 92-99.

Shneiderman, Ben, Dynamic queries for visual information seeking, *IEEE Software 11*, 6 (1994), 70-77.

Spence, Robert and Apperley, Mark, Data base navigation: An office environment for the professional, *Behaviour & Information Technology 1*, 1 (1982), 43-54.

Spoerri, Anselm, InfoCrystal: A visual tool for information retrieval & management, *Proc. ACM Conf on Information and Knowledge Management* (1993).

Tufte, Edward, *The Visual Display of Quantitative Information*, Graphics Press, Cheshire, CT (1983).

Tufte, Edward, *Envisioning Information*, Graphics Press, Cheshire, CT (1990).

Tweedie, Lisa, Spence, Robert, Dawkes, Huw, and Su, Hua, Externalising abstract mathematical models, *Proc. ACM CHI96 Conference: Human Factors in Computing Systems*, (1996), 406-412.

Williamson, Christopher, and Shneiderman, Ben, The Dynamic HomeFinder: Evaluating dynamic queries in a real-estate information exploration system, *Proc. ACM SIGIR'92 Conference*, (1992), 338-346. Reprinted in Shneiderman, B. (Editor), *Sparks of Innovation in Human-Computer Interaction*, Ablex Publishers, Norwood, NJ, (1993), 295-307.

Wise, James A., Thomas, James, J., Pennock, Kelly, Lantrip, David, Pottier, Marc, Schur, Anne, and Crow, Vern, Visualizing the non-visual: Spatial analysis and interaction with information from text documents, *Proc. IEEE Information Visualization '95*, (1995), 51-58.

Wurman, Richard Saul, *Information Anxiety*, Doubleday, New York (1989).

Young, Degi and Shneiderman, Ben, A graphical filter/flow model for boolean queries: An implementation and experiment, *Journal of the American Society for Information Science 44*, 6 (July 1993), 327-339.