Supplemental Navigational Systems: The Emergence of Interactive Visual Components

Arno Scharl[1]
Christian Bauer[2]

[1]Information Systems Department
Vienna University of Economics & BA, Austria
Email: *scharl@wu-wien.ac.at*

[2]Working Systems
Perth, Australia
Email: *Christian.Bauer@ws.com.au*

Abstract

In complex electronic environments it is becoming increasingly important for the user to be able to verify the relation between different virtual spaces by employing supplemental navigational systems such as indexes, overview diagrams, or guided tours. This paper introduces a set of techniques for the conceptual design of such systems, with special regard to the generation and transformation of explicit graphical representations of the hypertext's node and link structure. It concludes with possibilities for integrating the methods in adaptive ways and applying them to the interface representations of advanced Web information systems.

Keywords

Human/computer interaction, information visualization, hypertext navigation,
view types, view transformations

INTRODUCTION

Nowadays, there is little disagreement about the fact that information represents a key economic resource and an organizational asset critical to survival. In distributed hypertext environments such as the World Wide Web it can become difficult to identify and efficiently access this resource. At the same time, the ability to successfully assimilate and exploit advanced Web technologies ahead of the industry curve requires a set of both technical and managerial skills. Being familiar and generally trusted, the generic term *engineering* is used to signify an orderly development in a new field (e.g., software engineering, knowledge engineering, document engineering, and so forth). In the context of Web-based systems, it denotes a systematic application of scientific knowledge (i.e., methods, tools, and technical concepts) to development, deployment, operation, and maintenance of complex Web information systems that meet a particular set of technical, economic, and social objectives (IEEE, 1999, Deshpande et al., 1999, Murugesan et al., 1999). To overcome the difficulties mentioned above, designing and deploying an efficient navigational system is vital to successful Web engineering.

The *primary navigational system* of a hypertext includes contextual links embedded in the textual information of a document and local non-contextual links (all kinds of links independent from the content of a page such as a set of buttons or pop-up menus). In electronic environments, the user can also verify the relation between different virtual spaces by retracing her steps or employing *supplemental navigational systems* such as indexes, overview diagrams, or guided tours. Many forms of spatial metaphors common in the real world (navigation, net, maps, paths, trails, signposts) and spatial imagery (three-dimensionality, textual landscapes, topography) inspired the early designers of hypertext systems and are still implemented for reasons of comprehensibility (Wenz, 1997, 577, Wexelblat, 1999). While the following section highlights the basic characteristics of visual supplemental navigational systems, the succeeding sections 0 and 0 introduce a number of basic view types and common transformations associated with them. Section 0 then applies the introduced concepts to the World Wide Web and gives an example of a tightly coupled view. The example is based on the extended World Wide Web Design Technique (eW3DT), which represents a conceptual meta model that has been introduced previously (Scharl, 1998, Scharl, 1997).

SUPPLEMENTAL NAVIGATIONAL SYSTEMS

Supplemental navigational systems are more than mechanisms to navigate a virtual space; they are crucial textual elements themselves, replete with their own interpretive assumptions, emphases, and omissions (Burbules and Callister, 1996). They are used to locate and interpret a given item of information, providing full context by verifying the relation between different items and the virtual spaces surrounding them. Therefore, they should also include mechanisms to signify the user's current location and to retrace her individual steps.

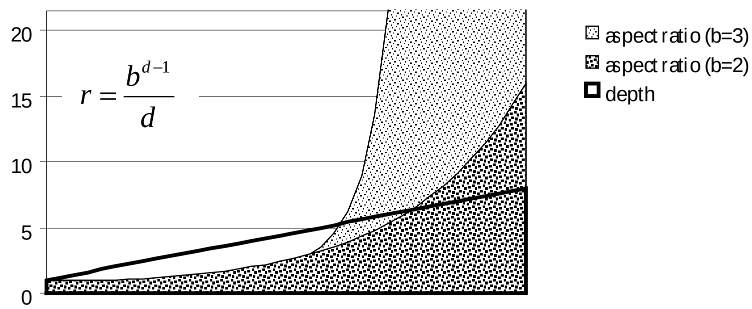
Hypertext systems represent semantic networks in themselves. Therefore, it is only natural to leverage the semantics that already exist in such networks (Kaplan et al., 1998, 48). A very intuitive way to tap the potential of such "hidden" semantics is the use of *site maps*, sometimes also referred to as *map windows*. In contrast to handcrafted representations of hypertext structures, automatically generated site maps are composed on the fly by the underlying system according to the system's topology and a set of pre-supplied layout rules (Andrews, 1995, 90). The underlying semantic relationship between information objects (= documents) is mapped onto a spatial arrangement, associated with the domain-dependent semantic properties of various links.

VIEW TYPES

Effective Web engineering is heavily reliant upon the provision of means for eliciting and reconciling the semantics of non-technical users (Hemingway, 1999, 284). Hierarchy represents an information artifact already familiar to every Web user. Cognitive scientists, however, disagree on the exact form of knowledge representation in human memory. Current models range from pictorially stored associations to complex propositional networks. Nevertheless, most experts agree on the information's structured and (where the domain permits it) hierarchical format (Mahling, 1994, 49f., Collins and Loftus, 1975, 407ff.). One of the first intellectual successes of infants, for example, consists of their mastery of the task of distinguishing material objects according to rather basic attributes such as shapes or colors. They gradually learn to compare and to notice similarities and differences (Tesch, 1990, 135). Since a hierarchical model is ingrained in the human brain to break down large sets of ideas into related categories, well-organized Web information systems most closely resemble this model (Rumpradit and Donnell, 1999). Hierarchies are almost ubiquitous and appear in many different locations, and are a great way to indicate prominence (Robertson et al., 1991, 189).

Thus it is not a coincidence that many decision support systems use hierarchical data sets (trees) to organize and represent complex sets of data points. Commonly used visualizations of these data sets include two- or three-dimensional node-link diagrams, space-filling tree maps, or tables of contents (Kumar et al., 1997, 103). Trees or directed graphs encode hierarchical data, typically visualized using connection (node-link diagrams, circular trees) or containment (tree maps, information slices). Whereas node positioning is irrelevant for the logical definition of a tree, it is important for its visualization. Unfortunately, large node-link diagrams acquire an extreme aspect ratio due to their exponential growth.

Figure 1 shows the truly remarkable dimensions of this growth, assuming squares with a side length of only 1.2 centimeters (drawn side by side) for calculating the drawing length of the last hierarchical level (Beaudoin et al., 1996, 87). The hypothetical 10x10 hierarchy would approximately cover the distance from Vienna to the Hawaiian Archipelago.



b	*d*	Nodes, total (#)	Nodes, last level (#)	Drawing length (km)
3	*3*	13	9	0.001
3	*6*	364	243	0.003
6	*6*	9,331	7,776	0.093
8	*9*	19,173,961	16,277,216	201.320
10	*10*	1,111,111,111	1,000,000,000	12,000.000

Figure 1: Aspect ratios of hierarchical trees with different branching factor (b=2, b=3), resulting in an exponential growth of fully populated node-link diagrams

As an inevitable result, these diagrams waste valuable screen space, especially when real names are used for identifying objects instead of acronyms and abbreviations. The same effect can be observed for outlines (Figure 2). Even a tree with a branching factor of only two, which is extremely unlikely for Web information systems, gets wider proportionally to $2^{(d-1)}$, resulting in an aspect ratio of $2^{(d-1)}/d$ (Robertson et al., 1991, 190). Therefore, static node-link diagrams of almost all but the most simple Web information systems come to resemble a straight line if not complemented by effective view transformations and interactive control.

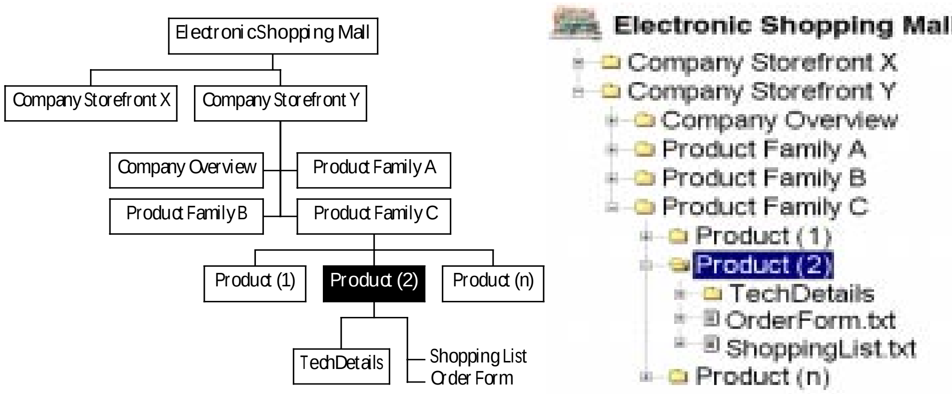


Figure 2: Hierarchical tree (left) and outline (right) representing an electronic shopping mall's simple hierarchical structure

A potential solution that addresses the problem of extreme aspect ratios is the rendering of cone trees, wrapping hierarchical trees into two- or three-dimensional displays. They dramatically increase the number of nodes that can be included in a square-shaped aspect ratio (Card et al., 1999, 149f.). Parent nodes in such a tree are located at the apex of the cone, while their children are arranged around the cone's circular base. Generated by the *Astra SiteManager (http://www.merc-int.com/products/astrasmguide.html),* the two-dimensional cone tree in the left part of Figure 3 visualizes publicly accessible nodes and links from the *University of California at Berkeley's* Web site (http://www.berkeley.edu/ as of August 14, 1999; only the first three levels of the hierarchical structure are considered). The recursive algorithm always uses the same circular layout, independent of an object's hierarchical position. Three-dimensional variations like the Euclidean cone tree depicted in the lower right corner of Figure 3 (Munzner and Buchard, 1995) are visually impressive but necessarily occlude some of the tree's objects, particularly in the case of dense information spaces. This negative effect can be lessened by interactively adjusting the user's viewing angle and position, or by transparently shading the body of each cone (Robertson et al., 1991, 190, Chuah et al., 1995, 65, Carpendale et al., 1997, 42).

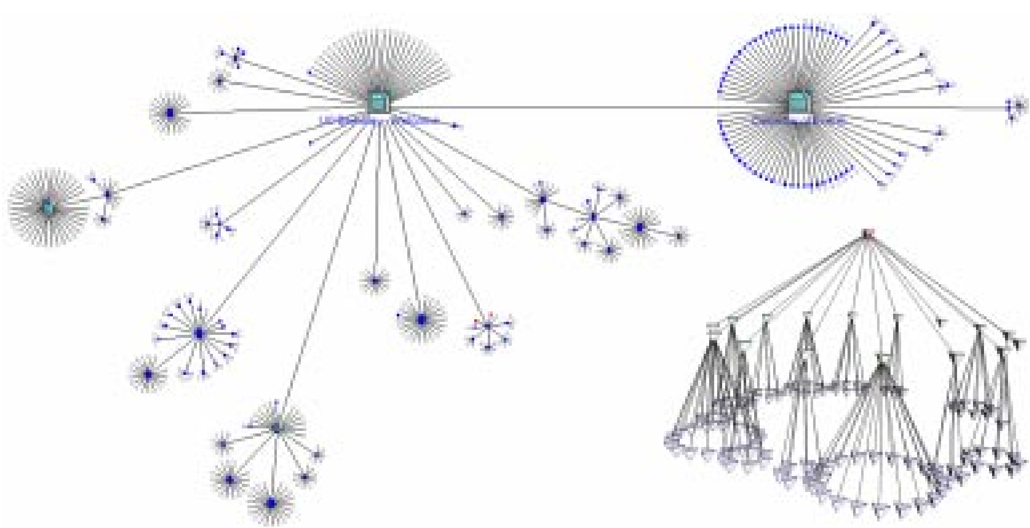


Figure 3: Two- and three-dimensional cone trees

Other (semi-)circular layouts for depicting structural hypertext features are *disk trees* and *information slices*, which depict the hierarchical levels as successive circles of increasing radius with a common center at the root of the tree (Chi et al., 1998, 404, Andrews, 1998, 41). Figure 4 contrasts both techniques while Figure 7 exemplifies the use of a disk tree as an overview diagram.

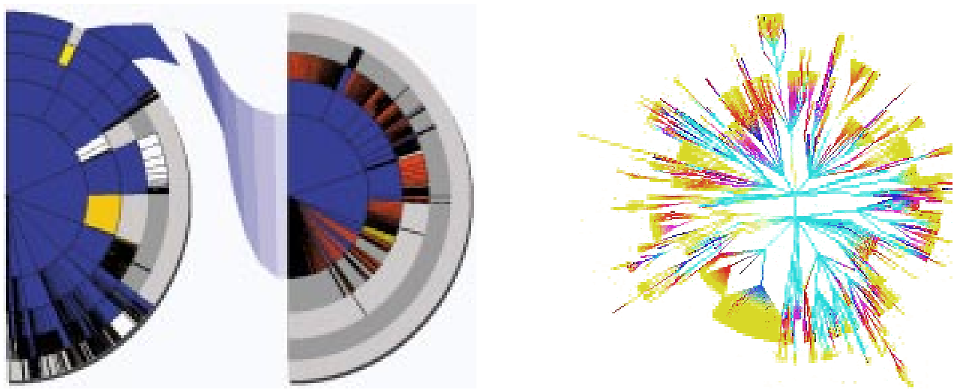


Figure 4: Information slices (left) and a disk tree (right) as examples of (semi-)circular visualization of hierarchical data (Chi, 1999b, 25, Chi et al., 1998, Andrews and Heidegger, 1998, Andrews, 1998, 41)

Unlike node-link diagrams, the concept of enclosure realized in tree maps by means of nested rectangles or in information slices by means of semicircular disks fills space rather than wasting it. Tree maps divide the space allocated for a node among its descendants according to certain numerically represented properties of the descendants. When the aspect ratio is held constant, both color and size of a rectangle can be used to compare particular target values. Thus it is relatively straightforward to realize the concept of adaptivity with the help of tree maps, as sections of the hierarchy relevant to the user’s current tasks and goals can dynamically be allocated more display size (Johnson and Shneiderman, 1991, 284). Unfortunately, tree maps tend to obscure the tree’s hierarchical structure and provide no way of focusing without losing context (Lamping and Rao, 1996, 37). Figure 5 compares a conceptual tree map representing the hierarchical structure already known from Figure 2 with a real-world example depicting more than 600 stocks of the New York Stock exchange as of August 12, 1999 (http://www.smartmoney.com/marketmap/).

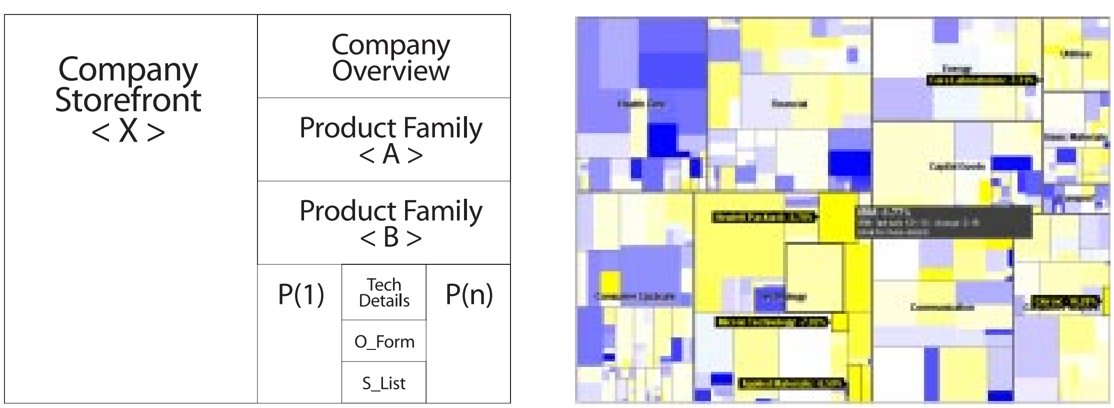


Figure 5: Conceptual versus real-world tree map

VIEW TRANSFORMATIONS

In almost all cases it is impossible to draw a diagram of the entire Web information system with its thousands of nodes and connecting links on a single computer screen. Therefore, view transformations that interactively modify and augment topographic maps on the basis of a number of structural or content parameters (e.g., depth bound, color mapping, annotations, and so forth) are crucial for the map's usability. Card et al. distinguish three common transformations (Card et al., 1999, 31):

- *Location probes* (also referred to as *brushing*) reveal the precise numerical or textual information for particular locations in the map on user's demand – e.g., pop-up dialog windows that give document details as the mouse is dragged over the document's representation in the display (Wright, 1995, 24, Johnson and Shneiderman, 1991, 288). They usually are integrated with one of the other two categories, viewpoint controls or distortions.
- *Viewpoint controls* determine the number of details shown by zooming, panning, and clipping the visual representation. The spatially indexed data can further be enriched by adding landmarks to provide rapid access (Card et al., 1999, 16). The main limitation of zooming is the lack of context as the details are magnified. To increase the number of variables per object shown for a given screen resolution, the number of objects or the quality of their representation necessarily have to be reduced. To overcome this intrinsic limitation, designers either implement multiple views or employ rapid and easy to invoke zoom algorithms.
- *Distortions* (e.g., *fisheye views*, *hyperbolic trees*, or *perspective walls*) show the entire information space in a single site map. This feature makes them better suited for certain applications. Their hyperbolic transformations shrink the nodes of the tree far from the root, thus balancing the needs for local detail and global context. For nautical navigation, compromised (distorted) projections of the round earth are also common. They enable graphical operations performed on the two-dimensional chart, with each type of projection sacrificing accuracy to support specific calculations (Card et al., 1999, 3f.).

IMPLICATIONS FOR HYPERTEXT NAVIGATION

Combining view types and transformations in various ways, two distinct approaches for providing visual site maps have become popular. The first one, *overview & detail*, integrates multiple views including at least one overview and one detailed view. The second one, *focus & context*, generates only one view that relies on distorted projections of the whole information space. In the latter case, the focal point can be either a particular node or simply a location in (eventually empty) space (Carpendale et al., 1997, 49). Figure 6 contrasts a file system's hyperbolic tree, generated by Inxight's interactive navigation tool *MagniFind (http://www.inxight.com/MagniFind_Download/)* with a conceptual screen layout based on multiple views (compare Figure 7).

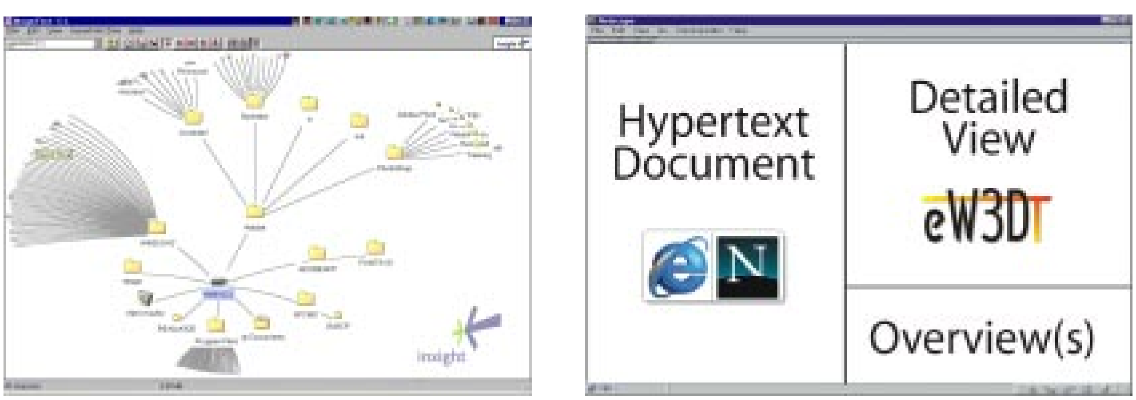


Figure 6: *Focus & context* versus *overview & detail*

Overview & Detail

When defining graphical excellence, Tufte recommends graphical displays that reveal the data at several levels of detail. He identifies layering and separation as the most powerful devices for reducing noise and enriching the content of displays (Tufte, 1983, 13, Tufte, 1990, 53). Site maps following this recommendation consist of two components that visually stratify the hierarchical document structure. The context-providing *overview* reduces search, allows the detection of global patterns, and aids the user in choosing the next node to visit. The *detailed view* displays a magnified focus for the local neighborhood surrounding the user’s current location (Nielsen, 1995, 3, Card et al., 1999, 285). To aid users in remembering their location, interface designers should provide some sort of "you are here" indicator as part of the overview (also referred to as *viewports* or *panners*). This "meta navigation" may be realized in the form of control widgets to pan the detailed view (e.g., sliders, buttons, or wire-frame boxes that can be dragged around in the overview). Similarly, scrolling the detailed view updates the position of the indicator in the overview. Hence, both views are said to be tightly coupled (Kumar et al., 1997, 104). The tight coupling of interface components, which is frequently complemented by additional zoom facilities to see more or less detail, comprises several conceptual aspects (Ahlberg and Shneiderman, 1994, 315f.):

- Continuous display of the information space with rapid, incremental and reversible interactions among components. This technique requires smooth direct manipulation operations such as zooming, rotation. and translation, which are computationally more demanding but cognitively more comprehensible (Card et al., 1999, 286, Munzner, 1998, 22).
- Intuitive and consistent mechanisms to guide users, progressively refine parameters, and show details on demand (often in combination with location probes; see above); Shneiderman, for example, recommends the following visual information seeking strategy: "Overview first, zoom and filter, then details-on-demand" (Shneiderman, 1997a, 23).
- Constraints on permissible operations to prevent errors and preserve logical display propositions.

Focus & Context

Distorted views following the focus & context approach provide great detail for those parts of the system in the vicinity of the user’s current location of interest and gradually diminishing amounts of detail for remote parts of the information space. As the nodes’ spatial positions and adjacency relationships often carry specific meaning, distorted views should not severely compromise them (Leung and Apperley, 1994, 139, Carpendale et al., 1997, 42). Meta navigation similar to the viewpoint controls' wire-frame box (e.g., direct manipulation by dragging a particular node to the center of the display) belongs to the common features of distorted views. It helps users perceive the larger, undistorted structure.

To enable distortions, the information space has to satisfy two conditions (Nielsen, 1995, 259f., Furnas, 1981): (a) the possibility to estimate the distance, either linear or structurally defined, between a given location and the user's current focus of interest, and (b) the existence of methods and tools to display the information at several levels of detail – e.g., variable resolution, highlighting, filtering, or selective aggregation (Card et al., 1999, 307f.). Hyperbolic projections are very common. Hyperbolic space, on which the hierarchies are laid out, is infinite in extent, just like Euclidean space. In contrast to Euclidean space, however, the circumference of a circle (and thus the available space for representing data points) grows exponentially with its radius. Therefore, hierarchical structures that also expand exponentially with depth can be conveniently mapped onto the display, representing a finite portion of Euclidean space (Lamping and Rao, 1996, 35f., Munzner, 1998, 20).

Integration via Tightly Coupled Views

Many of the geometrically complex displays provide only limited node and link semantics or no semantics at all. As additional context information quickly overwhelms the available display space, most of them rely on color coding or location probes to provide that information. This limitation makes them suitable for interactive navigation, but renders them practically useless for documenting or presenting Web information systems. The following sections introduce a concept based on multiple views that integrates methods from both categories, overview & detail and focus & context. It draws on the strengths and addresses the weaknesses of available view types and transformations to provide site maps that satisfy all the fundamental principles of interactive user interface design as postulated by Shneiderman (Shneiderman, 1997b, 74f., Bucy et al., 1999): strive for consistency, enable frequent users to access shortcuts, offer informative and intuitive feedback, permit easy reversal of actions, support an internal locus of control, and reduce short-term memory load.

Two-dimensional browsers such as Netscape *Navigator* or Microsoft *Internet Explorer* do not take advantage of the underlying structure of the information space. Figure 7 resolves that limitation by applying the concept of tightly coupled views introduced above to the electronic product catalog of Compaq as of January 1999 (http://www.compaq.com/showroom/). As Henry points out, there is no optimal layout that neglects the user's current region of interest (Henry and Hudson, 1991, 55ff., Henry, 1992, 81ff.). Tightly coupled views provide rapid access, and are simple to implement and understand. But their use frequently entails disruptive shifts of attention to the overview window.

Distorted views do not require such shifts, but face a number of other drawbacks (Card et al., 1999, 31f., Chuah et al., 1995, 69). Due to their geometrically complex transformations, distorted views can be more difficult to understand than tightly coupled views. Distant information may become so small that it cannot be perceived. For inexperienced users, the unlimited scope of most distortions may be disconcerting, because a local action may globally affect objects in the entire visualization in sometimes unpredictable ways. In distorted views based on hyperbolic geometry, for example, objects tend to get rotated during pure translations (Lamping and Rao, 1996, 45).

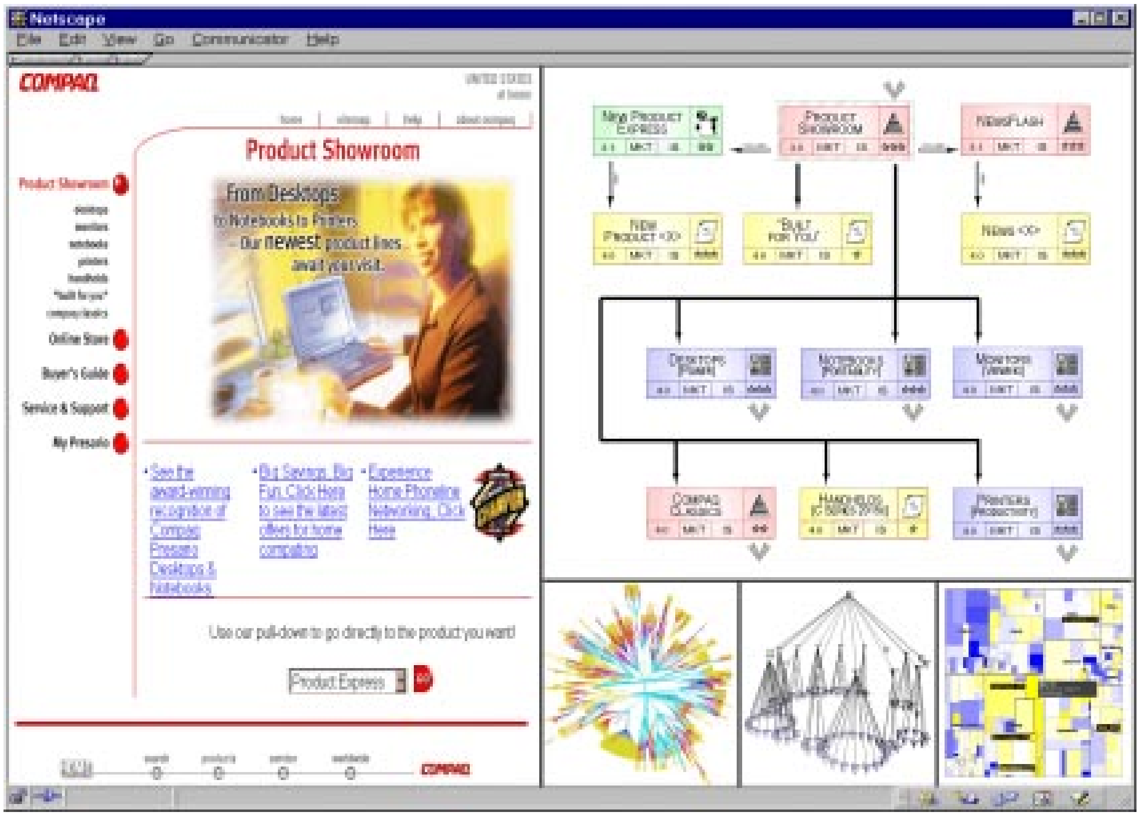


Figure 7: Example of tightly coupled view including a publicly accessible Web document, the corresponding eW3DT diagram, and three overviews

Site maps can show not only where users are located at the moment but also where they have already been during the current or preceding visits (their "footprints"). Furthermore, comparing the user's own path with the paths taken by others represents a special type of collaborative filtering, and may provide reassurance when implemented as a supplemental navigational system. Another promising method is to highlight nodes in the site map that contain relevant information. The two most common mechanisms are color coding to indicate a data value (e.g., closeness of the match regarding a full-text query) and contours to indicate a constant data value – e.g., visualizing iso-interest contours that connect nodes with the same hypothetical level of interest (Bryson et al., 1999, 85, Nielsen, 1995, 232, Furnas, 1981).

The extended World Wide Web Design Technique (eW3DT) uses point, line, and area marks to signify the topological structure of Web information systems, in most cases a directed graph (unlike their mathematical counterparts, point and line marks actually take up space - otherwise they would be invisible (Card et al., 1999, 28)). A detailed description of the eW3DT syntax and its application to Web information systems has been presented previously (Scharl, 1998, Scharl, 1997). eW3DT maps of complex Web information systems as part of high-level graphical interfaces reduce the cognitive overhead caused by a lack of spatial and temporal context (Turoff et al., 1999). They are an attempt to "build a visual interaction system that wastes the least amount of cognitive effort, thereby allowing users to direct the major part of their attention and cognitive processing power to the domain task" (Mahling, 1994, 42).

The right side of Figure 6 shows a conceptual screen layout including overview and an eW3DT map as detailed view on the upper right side. On the left side of the screen, the actual hypertext document is displayed. Figure 7 represents an application of this conceptual layout that uses three types of overview diagrams: disk tree, three-dimensional cone tree, and tree map (the overviews have symbolic character and do *not* represent the actual example). Of course, each of the other visualization techniques introduced in the previous sections such as the various types of hyperbolic projections, hierarchical trees, or outlines could be utilized as well. The user should be able to select type and arrangement of her preferred display techniques. The immediate access of full-screen versions of each display region at any point during the interaction blends *space multiplexing* (showing overview and detail displays at the same time in different parts of the screen) with *time multiplexing* (showing the various displays one at a time; (Card et al., 1999, 285)). Such a mechanism provides maximum flexibility and support of the user's current tasks and individual navigation style (Scharl, 2000). Besides adaptivity, the application of visual operators in spreadsheet-like applications represents another promising alternative to employing three different types of overview diagrams (Chi, 1999a, Levoy, 1994, 139ff., Hasler et al., 1994, 325ff., Furnas, 1981). Unlike numeric spreadsheets, which store only simple data elements and formulas in each cell, a cell in visualization spreadsheets can hold and manipulate abstract data sets of high complexity including appropriate selection criteria and exact viewing specifications (Chi, 1999b, 25).

CONCLUSION

Successful visualizations consist of "complex ideas communicated with clarity, precision, and efficiency" (Tufte, 1983, 51). However, it has been demonstrated above that a single layout algorithm cannot always produce the best results. Various conceptual techniques for designing intuitive interface representations of Web information systems have been presented, with special regard to the generation and transformation of explicit graphical representations of the hypertext's node and link structure. Web navigation frequently requires the user to make comparisons, or to look at some distant parts referenced from the current location. Consequently, representational adaptivity and multiple simultaneous foci (e.g., interactively animated overview diagrams) are essential keys to further improve the layout of supplemental navigational systems.

REFERENCES

Ahlberg, C. and Shneiderman, B. (1994) In *Conference on Human Factors in Computing Systems (CHI-94)* Association for Computing Machinery, Boston, USA, pp. 313-317.

Andrews, K. (1995) In *1st IEEE Symposium on Information Visualization (InfoVis-95)* IEEE Computer Society, Atlanta, USA, pp. 90-96.

Andrews, K. (1998) *IEEE Computer Graphics and Applications,* **18,** 40-42.

Andrews, K. and Heidegger, H. (1998) In *4th IEEE Symposium on Information Visualization (InfoVis-98)* IEEE Computer Society, Research Triangle Park, USA.

Beaudoin, L., Parent, M.-A. and Vroomen, L. C. (1996) In *2nd IEEE Symposium on Information Visualization (InfoVis-96)* IEEE Computer Society, San Francisco, USA, pp. 87-92.

Bryson, S., Kenwright, D., Cox, M., Ellsworth, D. and Haimes, R. (1999) *Communications of the ACM,* **42,** 83-90.

Bucy, E. P., Lang, A., Potter, R. F. and Grabe, M. E. (1999) In *49th Annual Conference of the International Communication Association (ICA-99)* San Francisco, USA.

Burbules, N. C. and Callister, T. A. (1996) *Educational Theory,* **5**.

Card, S. K., Mackinlay, J. D. and Shneiderman, B. (1999) *Readings in Information Visualization: Using Vision to Think,* Morgan Kaufmann, San Francisco.

Carpendale, M. S. T., Cowperthwaite, D. J. and Fracchia, F. D. (1997) *IEEE Computer Graphics and Applications,* **17,** 42-51.

Chi, E. H., Pitkow, J., Mackinlay, J., Pirolli, P., Gossweiler, R. and Card, S. K. (1998) In *Conference on Human Factors in Computing Systems (CHI-98)* Association for Computing Machinery, Los Angeles, USA, pp. 400-407.

Chi, E. H.-h. (1999a) In *Graduate School*University of Minnesota, Minneapolis, pp. 146.

Chi, E. H.-h. (1999b) In *Digital Library Workshop on Organizing Web Space (WOWS-99)* Association for Computing Machinery, Berkeley, USA, pp. 24-31.

Chuah, M. C., Roth, S. F., Mattis, J. and Kolojejchick, J. (1995) In *8th Annual Symposium on User Interface Software and Technology (UIST-95)* Association for Computing Machinery, Pittsburgh, USA, pp. 61-70.

Collins, A. M. and Loftus, E. F. (1975) *Psychological Review,* **82,** 407-428.

Deshpande, Y., Hansen, S. and Murugesan, S. (1999) In *First ICSE Workshop on Web Engineering (WebE-99)* (Eds, Murugesan, S. and Deshpande, Y.) Los Angeles, USA.

Furnas, G. W. (1981) Bell Laboratories, New Jersey.

Hasler, F., Palaniappan, K., Manyin, M. and Dodge, J. (1994) *Computers Physics,* **8,** 325-342.

Hemingway, C. J. (1999) In *IFIP WG 8.2 and 8.6 Joint Working Conference on Information Systems: Current Issues and Future Changes*(Eds, Larsen, T. J., Levine, L. and DeGross, J. I.) IFIP, Helsinki, Finland, pp. 275-286.

Henry, T. R. (1992) In *Department of Computer Science*University of Arizona, Tucson.

Henry, T. R. and Hudson, S. E. (1991) In *ACM Siggraph Symposium on User Interface Software and Technology*Association for Computing Machinery, Hilton Head, USA, pp. 55-64.

IEEE (1999) *Volume 1: Customer and Terminology Standards,* IEEE Computer Society Press, Piscataway.

Johnson, B. and Shneiderman, B. (1991) In *IEEE Visualization '91 Conference*IEEE, San Diego, USA, pp. 284-291.

Kaplan, C., Fenwick, J. and Chen, J. (1998) In *Adaptive Hypertext and Hypermedia*(Eds, Brusilovsky, P., Kobsa, A. and Vassileva, J.) Kluwer Academic Publishers, Dordrecht, pp. 45-69.

Kumar, H. P., Plaisant, C. and Shneiderman, B. (1997) *International Journal of Human-Computer Studies,* **46,** 103-124.

Lamping, J. and Rao, R. (1996) *Journal of Visual Languages and Computing,* **7,** 33-55.

Leung, Y. K. and Apperley, M. D. (1994) *ACM Transactions on Computer-Human Interaction,* **1,** 126-160.

Levoy, M. (1994) *Computer Graphics (Proceedings of ACM Siggraph-94),* **28,** 139-146.

Mahling, D. E. (1994) In *Cognitive Aspects of Visual Languages and Visual Interfaces*(Eds, Tauber, M. J., Mahling, D. E. and Arefi, F.) North-Holland, Amsterdam, pp. 41-75.

Munzner, T. (1998) *IEEE Computer Graphics and Applications,* **18,** 18-23.

Munzner, T. and Buchard, P. (1995) In *1st Annual Symposium on Virtual Reality Modeling (VRML-95)* San Diego, USA, pp. 33-38.

Murugesan, S., Deshpande, Y., Hansen, S. and Ginige, A. (1999) In *First ICSE Workshop on Web Engineering (WebE-99)* (Eds, Murugesan, S. and Deshpande, Y.) Los Angeles, USA.

Nielsen, J. (1995) *Multimedia and Hypertext: The Internet and Beyond*, AP Professional, Boston.

Robertson, G. G., Mackinlay, J. D. and Card, S. K. (1991) In *Conference on Human Factors in Computing Systems (CHI-91)* (Eds, Robertson, S. P., Olson, G. M. and Olson, J. S.) Association for Computing Machinery, New Orleans, USA, pp. 189-194.

Rumpradit, C. and Donnell, M. L. (1999) In *32nd Hawaii International Conference on System Sciences (HICSS-32)* IEEE Computer Society Press, Hawaii, USA.

Scharl, A. (1997) *Referenzmodellierung kommerzieller Masseninformationssysteme: Idealtypische Gestaltung von Informationsangeboten im World Wide Web am Beispiel der Branche Informationstechnik*, Peter Lang, Frankfurt.

Scharl, A. (1998) In *31st Hawaii International Conference on System Sciences (HICSS-31)*, Vol. IV: Internet and the Digital Economy IEEE Computer Society Press, Hawaii, USA, pp. 476-484.

Scharl, A. (2000) *Evolutionary Web Development*. London: Springer. http://webdev.wu-wien.ac.at/

Shneiderman, B. (1997a) *International Journal of Human-Computer Studies*, **47,** 5-29.

Shneiderman, B. (1997b) *Designing the User Interface: Strategies for Effective Human-Computer Interaction,* Addison-Wesley, Berkeley.

Tesch, R. (1990) *Qualitative Research: Analysis Types and Software Tools,* Falmer Press, New York.

Tufte, E. R. (1983) *The Visual Display of Quantitative Information,* Graphics Press, Cleshire.

Tufte, E. R. (1990) *Envisioning Information*, Graphics Press, Cleshire.

Turoff, M., Hiltz, S. R., Bieber, M., Fjermestad, J. and Rana, A. (1999) In *32nd Hawaii International Conference on System Sciences (HICSS-32)* IEEE Computer Society Press, Hawaii, USA.

Wenz, K. (1997) In *Semiotics of the Media: State of the Art, Projects, and Perspectives,* Vol. 127 (Ed, Nöth, W.) Mouton de Gruyter, Berlin, pp. 575-586.

Wexelblat, A. (1999) In *32nd Hawaii International Conference on System Sciences (HICSS-32)* IEEE Computer Society Press, Hawaii, USA.

Wright, W. (1995) In *1st IEEE Symposium on Information Visualization (InfoVis-95)* IEEE Computer Society, Atlanta, USA, pp. 19-25.

COPYRIGHT