Representation of Hierarchical Structures in 3D Space

André P. Calitz
Department of Computer Science and Information Systems, University of Port Elizabeth, P.O. Box 1600, Port Elizabeth 6000, South Africa. e-mail:csaapc@upe.ac.za
Donald Munro
Objecutive Inc., Fort Lee, New York, U.S.A. e-mail: donaldmunro@hotmail.com

Abstract

Hierarchical structures are found in much of the data that is used in everyday life including organisational management hierarchies and computer directory structures. The effective visualisation of large hierarchical information structures on limited size and resolution computer displays is important. The evolution of multimedia user interfaces is moving towards 3D graphical user interfaces for certain applications. One possible use of 3D on the WWW would be to extend the concept of presenting information of an organisation in three dimensions. The objective of this research project was to create an object-oriented framework that could be used to represent hierarchical data in 3D space using modified cone trees. Implementing and modelling the data used by a tertiary institution tested the framework. The research shows that frameworks can be used to represented hierarchical structures in 3D space and can assist users in manipulating and better visualising hierarchical data.

Keywords: Object-Oriented Framework, 3D Web Sites, Cone Tree, Hierarchical Structures.
Computing Review Categories: D.1.5, H.5, I.2.4, I.3.4.

1. INTRODUCTION

Information Visualisation (IV) entails presenting data to a user in order to derive knowledge and insight. Card et al [3] defines IV as "the use of computer-supported, interactive, visual representation of abstract data to amplify cognition". Cognition for a user is the acquisition or use of knowledge. Physical data, for example the earth's weather forecasts are shown on television in graphical format or an unborn child is scanned in the mother's womb and the data presented on the screen. Non-physical data, such as financial data and business information can also be presented in visual form, such as graphs, statistical techniques, diagrams, fish-eye techniques and using a Magic Lens [13].

Combining visualisation with information access techniques will assist the user in finding and relating to the information. In the past decade three-dimensional (3D) graphics software and hardware have advanced to the degree where semi-realistic views of real world or abstract concepts can be presented in real time. The availability of computer hardware and software supporting 3D graphics inevitably resulted in the technology also migrating to the World Wide Web (WWW).

Traditional computer user interfaces as well as user interfaces on Internet clients have evolved from simple text based command-line interfaces to the now familiar two-dimensional (2D) hypertext based graphical interfaces as seen on WWW browsers [14]. Human beings, however, tend to think and visualise in three dimensions. This is particularly true where the display of spatial information is concerned as it enables users to interact with the computer in the same way they interact with the physical world.

A user could therefore visualise and explore an organisation's data virtually and graphically. One possible use of 3D on the WWW would be to extend the concept of presenting information of an organisation in three dimensions. Information on an organisation would allow the visitor to explore the organisation's organisational hierarchy chart in 3D. The organisation hierarchy chart for a typical organisation might comprise the departments that belong to the organisation, the employees belonging to these departments and the functions fulfilled by the employees. To use the educational institution as an example, the institution could represent its hierarchical structures in 3 dimensions allowing users to easily navigate the hierarchy to obtain information on a particular item in the hierarchy such as a lecturer or a course given by a lecturer.

The development of these 3D web sites and the presentation of data can be seen as a specific application in the general problem domain of 3D web development. Object-Oriented Frameworks (OOF) provides a means of abstracting the common complexities of such problem domains and providing a 'semi-complete' application that the user can customise to produce the specific application that he/she requires [6].

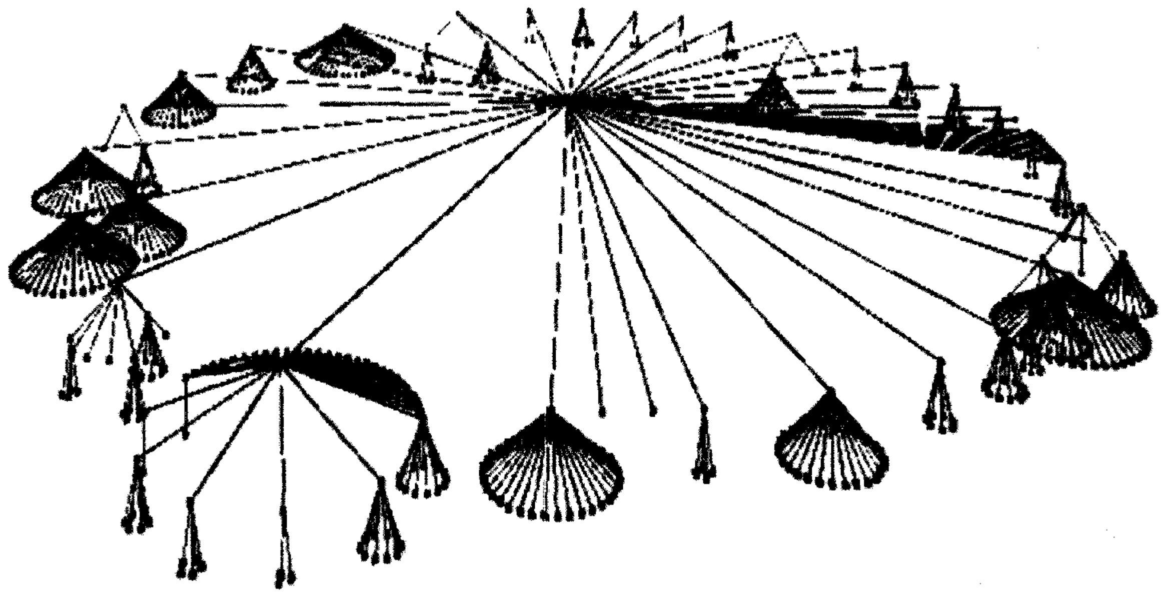

Figure 1: A Cone Tree

The development of an OOF for the 3D web development domain would thus be one way of simplifying and abstracting the process of creating and presenting data in 3D and would provide the further advantage of promoting code reuse.

The objective of this research project was to create an open-ended OOF to enable the creation and modelling of 3D web sites, representing the internal detail of buildings used by organisations as well as the hierarchical organisational structure. In order to describe the project, the conceptual basis for OOFs will be outlined. Methods for the representation of hierarchical structures will be discussed, focusing specifically on the use of cone trees for presenting hierarchical data. The design and implementation of hierarchical structures using modified cone trees will then be discussed followed by an outline of the testing procedures. This paper will only focus on using the OOF to present the hierarchical structure of an organisation in 3D space.

2. REPRESENTATION OF HIERARCHICAL STRUCTURES

Hierarchical structures are found in much of the data that is used in everyday life including organisational management hierarchies, document contents (chapters, sections, etc), computer filing systems (directory structures) and the structuring of pages on a WWW site. The effective visualisation of large hierarchical structures on limited size and resolution computer displays is important. When exhibiting such information in 2D, the visualisation space can become cluttered very quickly, even with moderate sized datasets. Visualising these hierarchical structures in 3D space eases the clutter problem although it does not completely solve it. Many users also find 3D hierarchical structures easier and more intuitive to navigate.

The cone tree was one of the first 3D hierarchical representations and was developed at Xerox PARC by Robertson et al [11]. A cone tree is a 3D representation of a tree structure, i.e. a standard G=(V,E) graph with vertices and edges. In a cone tree representation, the root of a tree (represented by a cube, a sphere or some other appropriate object) is located at the tip of a transparent cone. The children of the root node are arranged around the base of the cone. Each child can be the root node of a subtree, which is represented in a recursive fashion by a cone whose tip is located at the object representing the child (Figure 1). Cone tree visualisation can be improved, particularly for very large datasets, by techniques such as usage-based filtering, animated zooming, hand-coupled rotation, fish-eye zooming [7], coalescing of distant nodes, texturing, effective use of colour for depth cueing, the applications of dynamic queries [4] and node expansion-contraction [5].

The following are some other methods for the representation of hierarchical structures:

- The fish-eye technique as discussed by Furnas [7] has been applied to various representation methods other than cone trees. This technique determines the size and position of information by using a degree-of-interest (DOI) function. Usually the DOI function uses the distance from the focus and the importance of the information;
- The Perspective Wall shows a detailed area of the information space, in 2D [10]. The remaining information is presented in a 3D perspective. The 3D view allows more information to be displayed than a 2D view would allow;
- Magic Lens is a moveable lens the user can slide over the data [13]. The lens changes the display of information by applying a viewing operation to the underlying information.

Thus the lens can show different details within arbitrary regions while still preserving the context outside the regions;

- Variable Zoom uses hierarchical clustering in conjunction with a fish-eye [12]. Nodes are clustered using task-specific relationships. The clustering is hierarchical, thus users can choose at what level they wish to view the information;
- Hyperbolic transformations lay out the hierarchy onto a hyperbolic plane and then map the plane onto a circular display region [9];
- The Cheops method uses tessellation of triangles representing information nodes to compress the information space [1];
- Statistical methods have been used to analyse collections of data in an attempt to group objects together according to some measure of semantic "closeness" (i.e. do they logically belong together). The resulting proximity measures are typically scaled and returned as numerical values that are then used to cluster the objects in 3D space [2]; and
- A fractal-based hierarchical viewing method, utilising the self-similar geometrical characteristic of a fractal, that allows users to visually interact with a huge tree in the same manner at every level of the tree, has been developed [8]. The fractal dimension, a measure of complexity, makes it possible to control the total amount of displayed nodes.

3. OBJECT-ORIENTED FRAMEWORKS

The increase in software complexity over the past two decades has led to the processes of software design and development evolving from the era of procedural design into the current Object-Oriented Design (OOD) standards. Software applications are becoming bigger and more complex and user expectations are much higher than in the past. This has led to increases in the cost of software development and an increase in the number of failed projects. While the code reuse provided for in normal OOD has solved many problems, the sheer size of many applications has led to the requirement for new more application domain specific OOD methods and the OOF is one such solution.

A framework is a reusable, semi-complete application that can be specialised to produce custom applications [6]. The classes that make up the framework provide a skeleton that a developer can use and combine to build applications that fall within the application domain that the framework addresses. Due to the framework being the actual implementation of the application, the run-time architecture of a framework is characterised by an 'inversion of control' [6]. This means that in an application developed using a framework the main thread of control resides in the framework and not in code written by an application developer using the framework. When events occur, the framework's dispatcher reacts by invoking hook methods on application developer coded handler objects that perform application-specific processing on the events. A framework provides design reuse as it allows developers with less knowledge of the application domain covered by the framework to reuse code designed by developer(s) with an in-depth knowledge of the application domain.

The aim of the project was to build an OOF to enable the creation and modelling of 3D web sites representing the internal detail of buildings used by organisations as well as the hierarchical organisational structure within organisations. The user should be able to navigate around this 3D representation and enter and exit floors and rooms within the building. Additionally the user should be able to click on items of interest that occur inside the building to obtain more information or display a multimedia view (these features are not discussed in this paper). The framework further had to cater for the 3D display of hierarchical information pertaining to the organisation. Links to multimedia content was also supported within the framework. The framework had to be data driven, that is, all information used to create the detail in the world will be stored in a database which the framework accesses when rendering the world. The framework utilised industry standard technologies and be flexible enough to allow application developers to customise and reuse it through object-oriented constructs such as inheritance and polymorphism.

The framework specifically had to also allow for the modelling of the hierarchical associations that occur in departments and organisations such as linkages between departments and employees and offices and between employees and functions associated with the employee, e.g. courses presented by a lecturer. These hierarchical associations are represented in a 3D image through which the user can navigate. As the relationship between organisations, buildings and floors is also hierarchical, the hierarchical view can also be used for high-level navigation of these entities. For the purposes of this framework a modified version of the cone tree representation was considered appropriate as it provides an intuitive interface for the end user and it is relatively easy to provide a flexible configurable class hierarchy that can easily be customised by an application developer.

3.1 Cone tree Modifications and Implementation in the OOF

In the standard cone tree model clicking on a non-leaf node in the cone tree opens another cone tree below the clicked node with the clicked node being the root node for the new cone tree (Figure 1). The problem with this approach is that it quickly creates a cluttered user interface and requires small datasets and a high-resolution display to be practical. The framework design for the cone tree differs in two ways:

1. The cone tree works like a hypertext system in that clicking on a node redraws the cone tree with the clicked node as the new root node at the top of the display with all children nodes of that node in the hierarchy appearing as nodes in the cone tree. Clicking on the root node takes the end user back to the previous layer in the hierarchy. The advantages of this approach are:

- It is more space efficient as it does not fill the display with confusing detail of previous layers in the hierarchy; and
- Web users are very familiar with the hypertext metaphor for data exploration and would easily relate to data visualisation.

However, the disadvantage is that less information on hierarchical position is displayed.

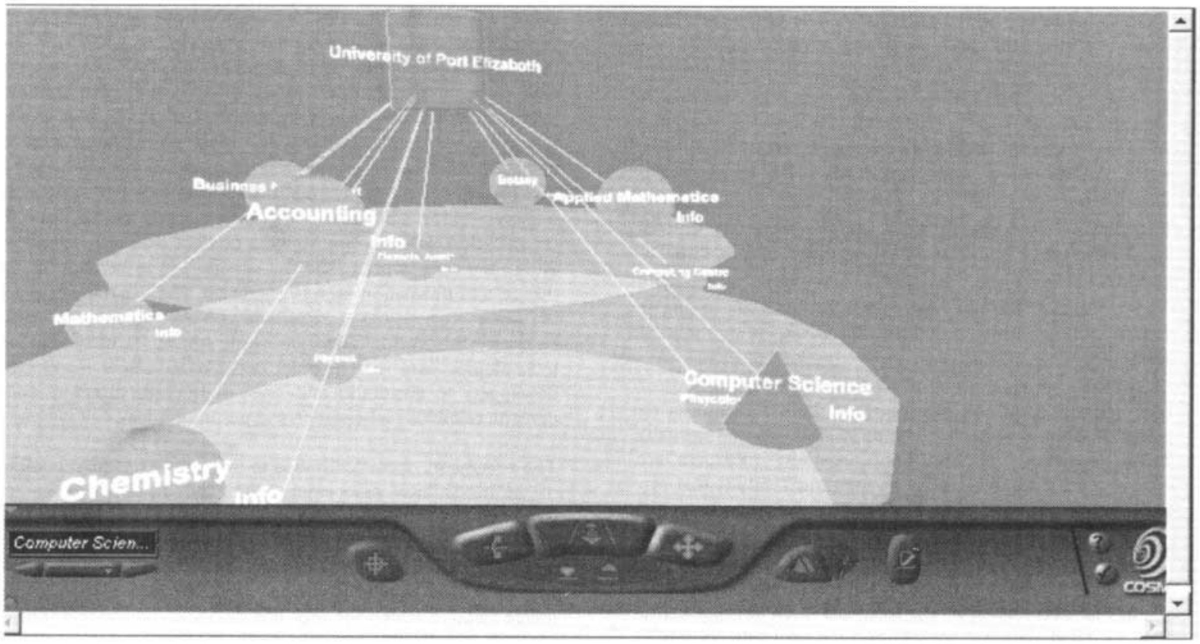

Figure 2: Example of a Cone Tree generated by theOOF

2. Because the cone tree only shows one level in the hierarchy at a time it does not need to display all the nodes for a hierarchy level on the same plane. Instead better use can be made of the display space available by providing more than one layers for hierarchy levels with large numbers of nodes with successively larger number of nodes on succeeding layers. Users can browse the data by moving up and down as well as in the x and y planes of 3D space, while the framework can provide quick navigation to particular nodes for users who know what they are looking for by providing a list of viewpoints for each node displayed that the user can select to jump to.

A cone tree, as displayed by the framework (Figure 2), consists of nodes representing objects in a level of the hierarchy being displayed with the root node being the parent node for that level. Each node actually represents a class in a hierarchy such as a Department or an Employee. In order to allow the cone tree classes to deal with these represented classes, a method is required which allows the cone tree classes to manipulate these represented classes generically.

Currently the only OO language capable of supporting web based development and interaction with other web technologies is Java, thus the framework was implemented using Java. Several 3D rendering technologies were identified including VRML, Java3D, OpenGL and DirectX that satisfied the main requirement of being model based and easily programmable. It was decided that only a VRML rendering package would be developed initially as VRML provides higher-level primitives and does much of the work that the framework rendering would have had to do had they been developed in Java3D, OpenGL or Direct3D.

The two main aims of the cone tree implementation are to add as much functionality as possible to the framework in an attempt to reduce the amount of work that the framework user needs to do in order to make a cone tree work, and to make the implementation as flexible as possible so as to allow users to configure and modify the behaviour of the cone tree as required. The cone tree implementation is a modified version of the classical cone tree in that the cone tree works like a hypertext system. Clicking on a node redraws the cone tree with the clicked node as the new root node at the top of the display with all children nodes of that node in the hierarchy appearing as nodes in the cone tree. Because only one level in the hierarchy is displayed at a time multiple layers of nodes are possible allowing more nodes to be displayed (Figure 2).

The user is allowed to insert as many links as required for each node in the cone tree. These links appear as text or icons next to the node and may be clicked on independently from the node to provide other functionality on a node-by-node basis. The framework optionally provides two links that are handled automatically. These links are the *Info* link, which, when clicked on can display multimedia information in a separate HTML frame on the node.

The *Go* link enables the view to be changed from the cone tree mode to the building interior mode. The *Go* link applies to nodes representing spatial objects such as buildings and floors. This allows the use of the cone tree as an abstract browsing view for spatial objects. The ConeTree and ConeTreeNode classes provide several ways to customise the appearance of the cone tree. The shape, size and colour of the three types of nodes (root, leaf and normal) can be specified either for the entire cone tree or at individual node level. The application developer can also specify whether to draw lines connecting the root node with the subordinate nodes and the colour of these lines.

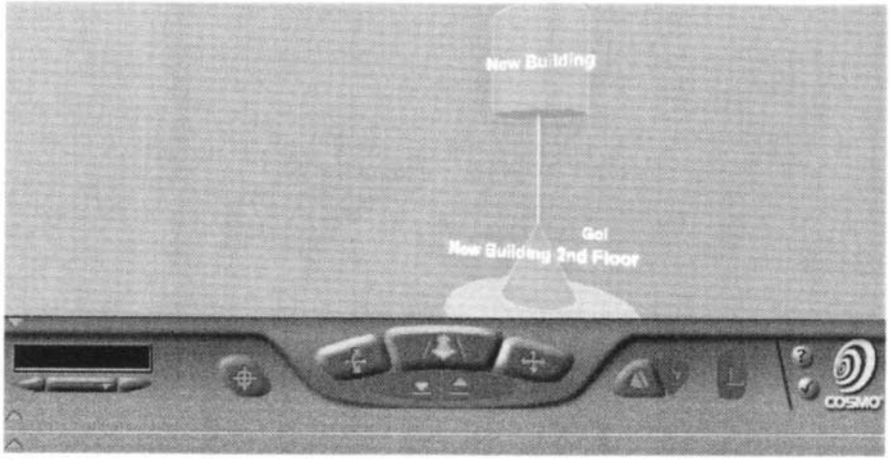

Figure 3: Selecting a floor using the hierarchical view

4. EVALUATING THE OOF CONE TREE IMPLEMENTATION

In order to test and evaluate the framework two floors of the University of Port Elizabeth (UPE) were implemented. The UPE Department of Computer Science and Information Systems (CS/IS) floor is a relatively large floor with more than 40 rooms and is utilised mostly for educational purposes. It also provides office space for departmental staff, while the UPE Marketing Department floor is smaller and bears a closer resemblance to a floor and department of a non-educational organisation as it supplies a non-academic service to the organisation. The data describing the spatial definition of the floors was entered into the database. The various room content items such as furniture and textures were also entered in the database along with any links to multimedia data. The organisational structure for the departments, that is, the linkages between departments, employees and employee functions were entered to test the organisational hierarchical view implemented by the framework.

A sample application was then created using the framework to allow the user to navigate through the spatial model held in the database. The sample was designed to allow the user to browse the hierarchy of building and floors using a cone tree hierarchical view to select the floor or building he would like to display (Figure 3). Additionally the user could switch to an organisational hierarchical view to browse a cone tree representation of the organisational hierarchy displaying information on the relationships between organisations and departments, departments and employees and employees and their functions (Figure 4). The user could also click a Go icon on some of these nodes that had links back to building structures and the application would display the building view mode for this link. For example clicking a Go node for an employee would redraw the display in the spatial 3D mode in the employee's office.

Several users were initially requested to evaluate the implementation and interviewed after navigating through the data. Users found the presentation of data and the specific use and implementation of cone trees easy to navigate and they could relate to the visualisation of the data. Users further indicated that they could relate with ease to the presentation of hierarchical information. Usability testing in the new UPE Usability Laboratory of the cone tree implementation is currently been undertaken in order to compare the implementation with existing hierarchical information visualisation techniques.

5. SUMMARY

Information visualisation can assist users in manipulating and better understanding data and specifically hierarchical data relating to organisations. The Internet has provided users access to information presented mainly in a 2D format. The current trend in multimedia applications is towards a 3D interface for accessing and presenting information however this requires more complex software. The increase in software complexity has led to new object oriented design standards and methods. The framework allows for the 3D representation of the internal structure and contents of multiple buildings associated with an organisation and hierarchical data of the organisation. The OOF has been tested and showed the practical use of such a domain specific application for a tertiary institution. Cone trees were selected as the visualisation technique used to present hierarchical data.

The objective of this research study was to create a data driven OOF that could be used to develop 3D web sites in order to depict both spatial and abstract representations of organisations. An adapted cone tree was implemented and found to be more space efficient, as it does not fill the screen with confusing detail from previous layers in the hierarchy. Users, who were familiar with the hypertext metaphor to do data exploration, found it easy to use the cone tree implemented for the visualisation of organisation hierarchical data.

The framework can be downloaded from http://www.ecp.co.za/~DonaldM/TdOfc/Default.html.

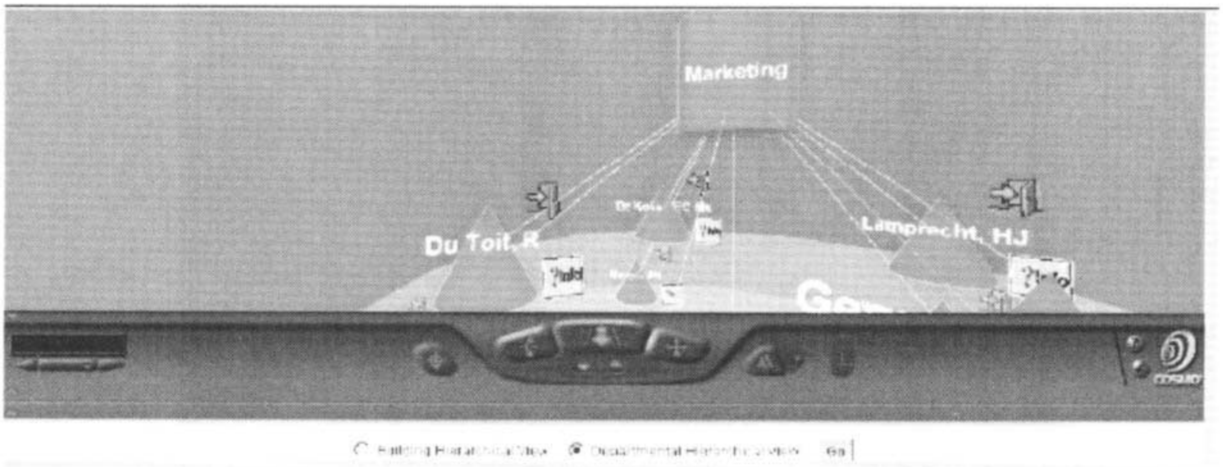

Figure 4: Hierarchical display of Marketing Department Information

References

[1] L. Beaudoin, M.A. Parent, and L.C. Vroomen. Cheops: A Compact Explorer For Complex Hierarchies. *Proceedings of Visualization '96*, 1996.

[2] S. Benford, D. Snowdon, C. Greenhalgh, R. Ingram, I. Knox, and C. Brown. VR-VIBE: A Virtual Environment for Co-operative Information Retrieval. *Eurographics'95*, August 1995.

[3] S.K. Card, J.D. Mackinlay, and B. Shneiderman, Readings in Information Visualization using Vision to Think. Morgan Kaufmann Publishers, Inc. 1999.

[4] J. Carrièrre and R. Kazman. Research Report Interacting with Huge Hierarchies: Beyond Cone Trees. *Proceedings of Information Visualization'95*, October 1995.

[5] R. Chimera, K. Wolman, S. Mark, and B. Shneiderman. Evaluation of three interfaces for browsing hierarchical tables of contents. *Technical Report CAR-TR-539, CS-TR-2620.* University of Maryland, February 1991.

[6] M.E. Fayad and D. Schmidt. Object-Oriented Application Frameworks. *Communications of the ACM, Vol. 40, No. 10,* http://siesta.cs.wustl.edu/~schmidt/CACM-frameworks.html, October 1997.

[7] G.W. Furnas. Generalized fisheye views. *Proceedings of CHI'86 Human Factors in Computing Systems*, 1986.

[8] H. Koike and H. Yoshihara. Fractal approaches for visualizing huge hierarchies. *Proceedings of the 1993 IEEE Symposium on Visual Languages*. IEEE, 1993.

[9] J. Lamping, R. Rao, and P. Pirolli. A Focus+Context Technique Based Geometry for Visualizing Large Hierarchies. *Proceedings of CHI'95 Human Factors in Computing Systems*, 1995.

[10] J. Mackinlay, S. Card, and G. Robertson. Perspective Wall: Detail and Context Smoothly Integrated, *Proceedings of the ACM SIGCHI'91 Conference on Human Factors in Computing Systems*, 1991.

[11] G. Robertson, J. Mackinlay, and S. Card. Cone Trees: Animated 3D Visualizations of Hierarchical Information. *SIGCHI '91 Proceedings*, 1991.

[12] D. Schaffer, Z. Zuo, L. Bartram, and J. Dill. Comparing Fisheye and Full-Zoom Techniques for Navigation of Hierarchically Clustered Networks. *Proceedings of Graphics Interface*, 1993.

[13] M.C. Stone, K. Fishkin, and E.A. Bier. The Movable Filter as a User Interface Tool. *Proceedings of CHI'94 Conference on Human Factors in Computing Systems*, 1994.

[14] W3C World Wide Web Consortium. About The World Wide Web. *http://www.w3.org/WWW/*, 1999.