Late Breaking Hot Topic Paper, IEEE Symposium on Information Visualization (InfoVis'98), Research Triangle Park, North Carolina, Oct. 1998.

Information Slices: Visualising and Exploring Large Hierarchies using Cascading, Semi-Circular Discs

Keith Andrews Helmut Heidegger *

IICM, Graz University of Technology, Austria

ABSTRACT

This paper presents work in progress on a new technique for visualising and manipulating large hierarchies. The information slices approach compactly visualises hierarchical structures using a series of semi-circular discs. The technique is described in the context of our early experience with a prototype file system visualiser based on information slices.

Keywords: Information visualisation, hierarchies, trees, discs.

1 INTRODUCTION

Over the last few years, the emerging field of information visualisation has resulted in a numerous techniques for helping visualise and make sense of large information spaces [10]. Among these are several techniques for visualising and interacting with large hierarchies which go beyond the traditional approach taken in 2d scrolling browsers such as the Windows Explorer.

Treemaps [6, 9] are space-filling visualisations of hierarchies based on successive horizontal and vertical subdivision of screen rectangles. The area of each rectangle is proportional to some attribute of the underlying hierarchy such as the (total) size of each subtree.

Xdu [4] is a utility for the X window system which displays a graphical disk usage for Unix file systems. Rectangles are stacked from left to right as the directory tree is descended. The current directory is represented by the leftmost rectangle, which is the entire height of the window. Subdirectories are represented by neighbouring rectangles in the next column, whose height are proportional to the size of each subdirectory.

The hyperbolic browser [7] uses a focus and context technique based on hyperbolic geometry. A hierarchy is laid out uniformly on the hyperbolic plane and then mapped to the unit disc for display on screen. Nodes in the centre of the disc are largest and nodes are assigned progressively less space towards the perimeter of the disc.

Cheops [3] is based on multiple re-use of overlaid triangles in the display. Working top-down, the selection of a node (triangle) at a particular level designates that node's children are to be represented by the next lower level of triangles.

Other techniques such as cone trees [8], FSN [11], the Harmony information landscape [1], and information pyramids [2] utilise three-dimensional space to compactly represent large hierarchical structures.

2 INFORMATION SLICES

The information slices technique uses one or more semi-circular discs to compactly visualise large hierarchies in two-dimensional space. Each disc represents multiple levels of a hierarchy; typically between 5 and 10 levels are visualised on each disc, the exact number being user configurable. Deeper hierarchies are represented using a cascading series of discs. At each level of the hierarchy, the children are fanned out in the available space according to the total size of each child.

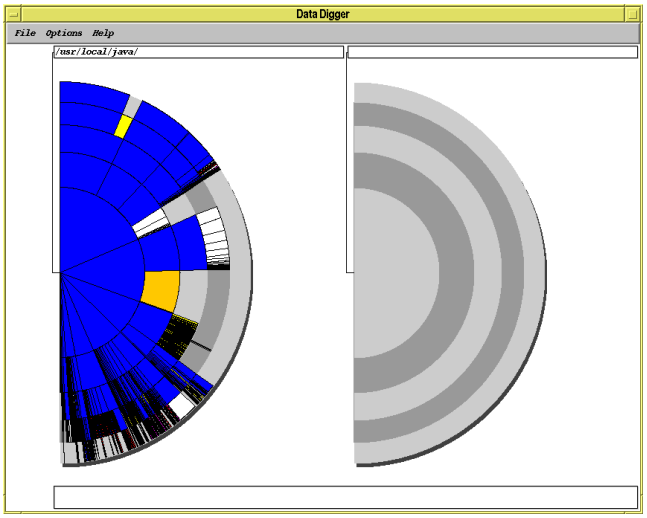

Figure 1: Information slices visualisation of the JDK 1.1.6 distribution, showing the root directory and five levels of its subdirectories.

We have implemented a prototype of information slices for visualising the hierarchical tree structure of directories and files in a file system. Figure 1 shows the information slices representation of Sun Microsystem's JDK 1.1.6 distribution for Solaris, which has 6158 files, arranged in 502 directories, with a maximum depth of 9. The lefthand disc shows the top 5 levels of the JDK hierarchy, whilst the righthand disc is as yet unused.

Figure 2 shows the display after the user has chosen to expand the `swing/com/sun/java/swing` subdirectory into the righthand disc. Directories are coloured blue and other files are colour-coded according to their type.

Further expanding the `text/html` subdirectory causes the leftmost disc to be iconified and slide off the screen to the left, the right disc slides over to the left, and a new disc is opened up on the right, as shown in Figure 3. Such cascading of discs supports rapid exploration of very deep hierarchies. An iconified disc can be restored by clicking upon it.

The user can interactively configure how many levels of hierarchy should be displayed on each disc, using the Options panel shown in Figure 4. For example, in Figure 5, the user has selected 8 levels of hierarchy to be displayed in each disk.

By default, the space allocated to a directory's slice is propor-

* kandrews@iicm.edu, hheide@iicm.edu
IICM, Schießstattgasse 4a, A-8010 Graz, Austria.

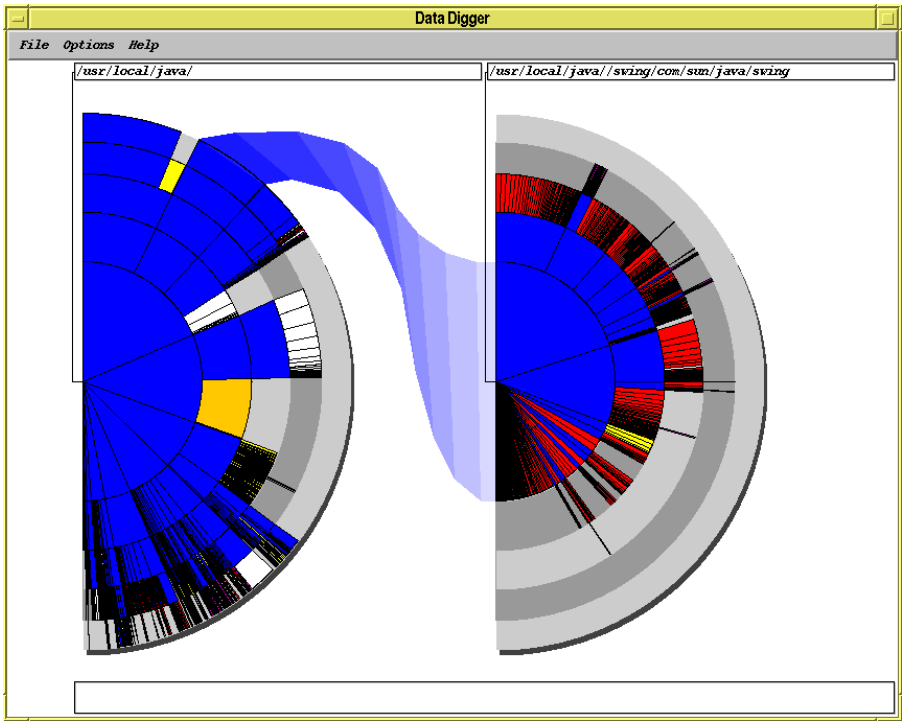

Figure 2: Expanding the `swing/com/sun/java/swing` subdirectory into the righthand disc.

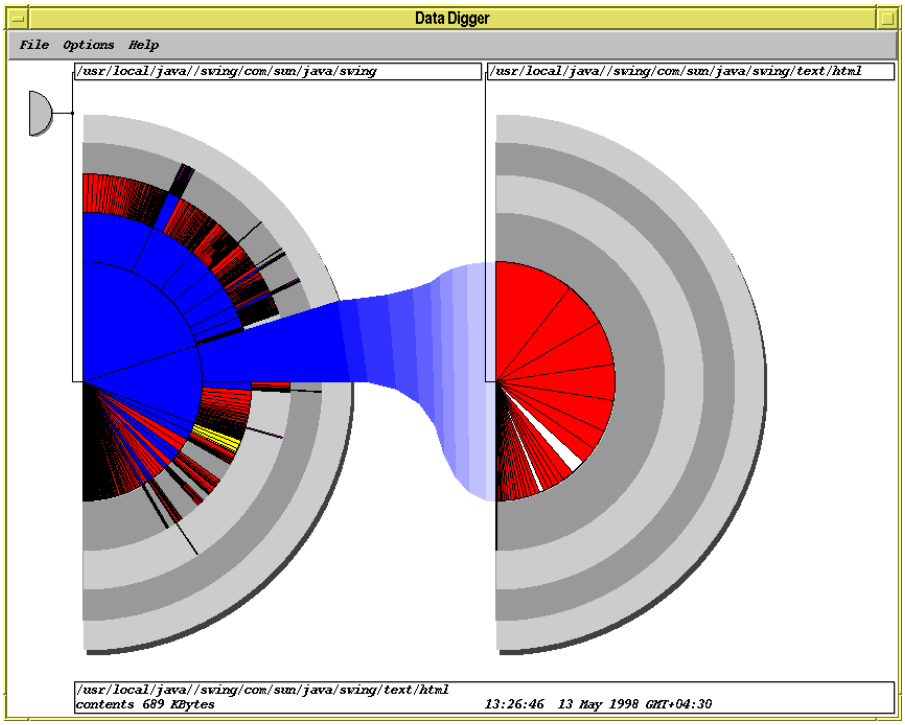

Figure 3: Further expanding the `text/html` subdirectory causes the lefthand disc to be iconified off to the left of the screen and a new disc to be opened to the right.

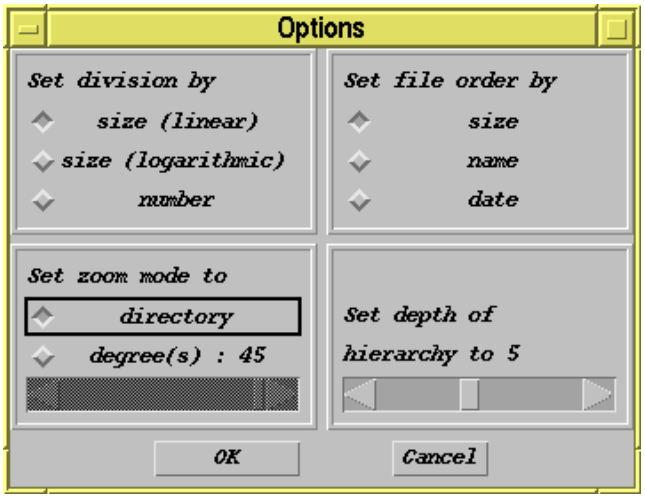

Figure 4: The Options panel allows various setting to be configured by the user interactively.

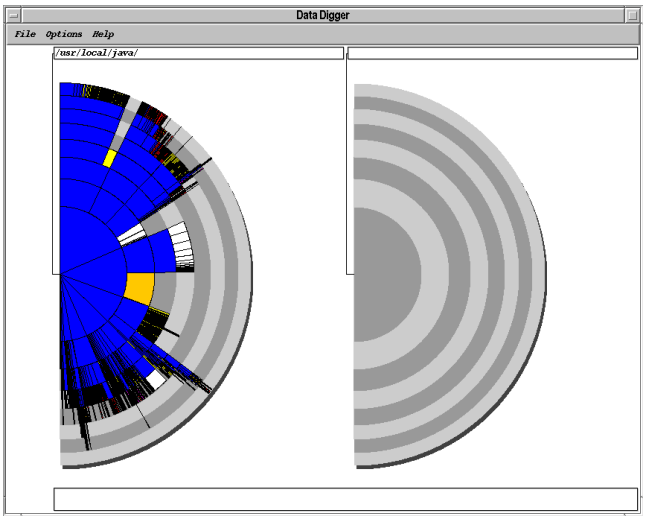

Figure 5: The JDK distribution shown on a single disc of 8 levels.

tional to the total size in bytes of files belonging to it and all of its subdirectories. The Options panel allows the user to change this mapping to, for example, the total number of files contained in a branch of the tree. Furthermore, the order of listing each child from top to bottom can be changed from descending order of size (the default) to alphabetical by name or chronological by modification time. Figure 6 shows the JDK 1.1.6 distribution, where the size of each slice is determined by the total number of files, and children are listed alphabetically by name.

3 EARLY EXPERIENCE AND CURRENT WORK

The current prototype file system visualiser using information slices is written in Java and uses Java's Swing GUI components.

From our early experience, the information slices technique appears to be particularly well-suited to the rapid navigation of deep hierarchies. It is very easy to rapidly traverse many levels of a hierarchy. It is also easy to gain an overview of the relative sizes of parts of a tree.

Broad hierarchies can result in dense, thin slices, which are sometimes initially overwhelming. However, users can select particular (dense) slices of interest and fan them out in 180 degrees of their own in the righthand disc. We are considering providing some form of "handles" for interactive fanout of selected slices.

Currently, work is underway to integrate an information slices disc with a conventional tree explorer (the JTree component of Swing), so as to combine the advantages of both techniques. Users will hopefully be able to gain an overview and rapidly traverse in the disc, and then explore in detail in the explorer.

As space permits, the names of files and directories will also be displayed in the corresponding sector of the disc, in addition to the status bar.

Although the example application of information slices presented in this paper refers to the tree structure of files and directories in a file system, the technique is applicable to any kind of hierarchical structure. For example, the Hyperwave Information Server [5] supports explicit hierarchical structuring of the documents it manages into collections and subcollections. We intend to implement a visualisation and navigation facility for a Hyperwave web site's collection structure based on information slices.

4 CONCLUDING REMARKS

We have presented the information slices technique for visualising and interacting with large hierarchies. This work is still very much "in progress", but we believe that it will become a valuable addition to the palette of emergent tools for exploring large hierarchies.

This paper is available online with colour figures at `http://www.iicm.edu/keith/papers/ivis98/`

5 ACKNOWLEDGEMENTS

We would like to thank our colleagues at the IICM for their support and suggestions during this work and the anonymous reviewers for their helpful comments.

REFERENCES

[1] Keith Andrews. Visualising cyberspace: Information visualisation in the Harmony internet browser. In *Proc. First IEEE Symposium on Information Visualization (InfoVis'95)*, pages 97–104, Atlanta, Georgia, October 1995. `ftp://ftp.iicm.edu/pub/papers/ivis95.pdf`.

[2] Keith Andrews, Josef Wolte, and Michael Pichler. Information pyramids: A new approach to visualising large hierarchies. In *IEEE Visualization'97, Late Breaking Hot Topics Proc.*, pages 49–52, Phoenix, Arizona, October 1997. `ftp://ftp.iicm.edu/pub/papers/vis97.pdf`.

[3] Luc Beaudoin, Marc-Antoine Parent, and Loius C. Vroomen. Cheops: A compact explorer for complex hierarchies. In *Proc. Visualization'96*, pages 87–92, San Francisco, California, October 1996. IEEE Computer Society. `http://www.crim.ca/hci/cheops/paper.html`.

[4] Phil Dykstra. Xdu. Army Research Laboratory. `ftp://ftp.arl.mil/pub/xdu-3.0.tar.Z`.

[5] Hyperwave. `http://www.hyperwave.com/`.

[6] Brian Johnson and Ben Shneiderman. Tree-maps: A space-filling approach to the visualization of hierarchical

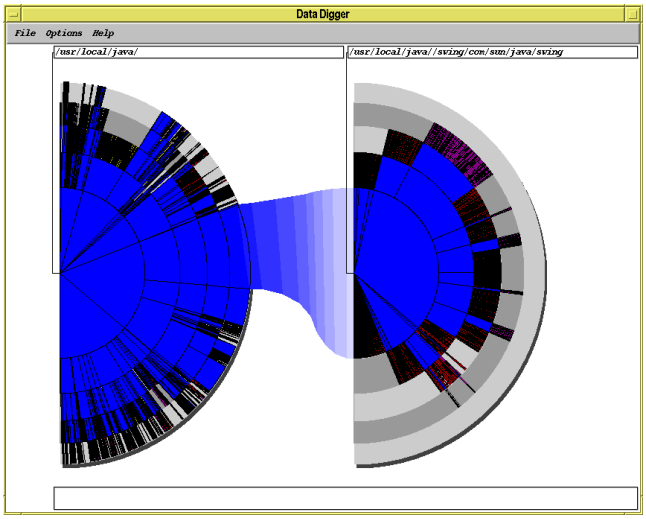

Figure 6: The JDK distribution, where the size of each slice is determined by the total number of files, and children are listed alphabetically by name.

information structures. In *Proc. IEEE Visualization '91*, pages 284–291, San Diego, California, October 1991. IEEE Computer Society. `ftp://ftp.cs.umd.edu/pub/papers/papers/2657/2657.ps.Z`.

[7] John Lamping, Ramana Rao, and Peter Pirolli. A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In *Proc. CHI'95*, pages 401–408, Denver, Colorado, May 1995. ACM. `http://www.acm.org/sigchi/chi95/Electronic/documnts/papers/jl_bdy.htm`.

[8] George G. Robertson, Jock D. Mackinlay, and Stuart K. Card. Cone trees: Animated 3D visualizations of hierarchical information. In *Proc. CHI'91*, pages 189–194, New Orleans, Louisiana, May 1991. ACM.

[9] Ben Shneiderman. Tree visualization with tree-maps: A 2-d space-filling approach. *ACM Transactions on Graphics*, 11(1):92–99, January 1992. `ftp://ftp.cs.umd.edu/pub/papers/papers/2645/2645.ps.Z`.

[10] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proc. 1996 IEEE Symposium on Visual Languages*, pages 336–343, Boulder, Colorado, September 1996. IEEE Computer Society. `ftp://ftp.cs.umd.edu/pub/papers/papers/3665/3665.ps.Z`.

[11] Joel D. Tesler and Steven L. Strasnick. Fsn: The 3d file system navigator. Silicon Graphics, Inc., 1992. `ftp://sgi.sgi.com/sgi/fsn`.