Modifiable Treemaps Containing Variable-Shaped Units

Frédéric Vernier and Laurence Nigay *

CLIPS-IMAG Laboratory - University of Grenoble

Abstract

This paper presents a novel method, for the visualization of hierarchical information such as directory structures or family trees. Our technique is based on nested Treemaps and inherits the main property of the Treemaps: space-filling display of hierarchical information. For each node, our technique allocates a bounding box according to a ratio (height/width) that can be modified by the user according to her/his task. To do so, our algorithm does not apply the slice and dice (top-down) approach of standard Treemaps. Our technique is effective, easy to implement and has a wide applicability. To illustrate its application, we developed a system called PARENT which displays a directory tree. PARENT is tightly coupled with Microsoft Windows Explorer.

CR Categories and Subject Descriptors: I.3.3 [Computer Graphics]: Picture/Image Generation- Display Algorithms; I.3.6 [Computer Graphics]: Methodology and Techniques - Interaction Techniques.

Additional Keywords: Visualization, Hierarchical Information, Treemap.

1 MOTIVATION

We present a new visualization method of large hierarchical structures. The method is based on nested Treemaps [3] but we do not apply the top-down (or dice-slice) approach of the classical Treemap algorithm. For each node, our method allocates a bounding box according to a ratio (height/width) that can be modified by the user. For example if the ratio equals 1, the corresponding visualization will contain bounding boxes that tend to be squares.

Instead the Treemap visualization [3] portrays each node as a rectangle. This visualization technique may not be appropriate for certain user tasks or for certain kind of information:

* BP 53, 38041 Grenoble cedex 9 France {frederic.vernier, laurence.nigay}@imag.fr

- Long but thin rectangles are difficult to select. The Fitts' law [2] confirms this.
- The surfaces of different rectangles are difficult to compare. As a proof, let us consider the following two rectangles: _______ >? _____. The task becomes even more difficult if we try to compare a horizontal rectangle with a vertical one.
- Vertical rectangles raise the problem of labeling. Figure 4 of the color plate, showing a standard nested Treemap, contains many long vertical rectangles in which it is impossible to display a label.

We consequently propose a new visualization method that lets the user select the shape of the nodes bounding box according to her/his task by designating a ratio.

2 ALGORITHM

Our algorithm consists of adding a new set of recursive steps between two recursive steps of the nested Treemap algorithm. One step re-arranges a set of rectangles into two sets of shapes close to the ratio defined by the user. As illustrated in Figure 1, the resulting tiled layout has exactly the same size as the initial set of rectangles.



Figure 1: One step of the recursive algorithm.

Based on the initial set of rectangles, our algorithm generates several options. Each option is generated by splitting the initial set into sub-sets. Numerous options are consequently generated because each sub-set is sliced horizontally and vertically. This aspect of the algorithm is not well illustrated in Figure 1 that shows only three options. The best candidate according to the fixed ratio is then selected. The evaluation function is very simple: A rectangle is represented by the ratio: height/width ($\Box = 1/2$ $\Box = 1/4$). For each option, the algorithm

computes the sum (named Eval in Figure 1) of differences between the fixed ratio and the ratio of each contained rectangle. The option that has the smallest sum is selected as shown in Figure 1. New sets of rectangles, such as the two sets in Figure 1, are then recursively re-arranged, until single rectangles are obtained. Once the algorithm ends, the generated patchwork of rectangles fills the same space as the initial set of rectangles.

The partitioning of the display space obtained by applying our algorithm inherits the four properties of the Treemaps [3]. One drawback of our algorithm is that the modification of the display of the full hierarchy due to the variation of one leaf can be higher than with standard Treemaps. Additionally, we experimentally observed that when a node has numerous sub-nodes, the obtained layout contains shapes close to the fixed ratio.

3 WORKING EXAMPLE: PARENT

We used our algorithm to design a system called PARENT which displays a directory tree. As shown in Figure 2 of the color plate, each folder and file occupies a display strictly proportional to its disk usage. Folders are displayed as windows and files as boxes.

Alternative views can be applied for the display of a folder. The alternative view mechanism consists of the possibility to replace the graphical layout of a folder by another view such as outlines or tree diagrams. PARENT implements only one alternative view similar to one of the views defined by Microsoft Windows Explorer, as illustrated in Figure 2 of the color plate: The alternative view is applied for the display of three folders.

A variety of display properties can be applied for conveying the characteristics of a file. Visual display properties and in particular color is one of the most important of these properties [3]. In PARENT two characteristics of the files are visually presented using color: the type and the last modification date.

- The type of a file is a non-ordered variable. We used the hue of the color to convey the type. Figure 3 of the color plate illustrates the use of eight different hues for conveying eight file types. The legend at the bottom of the display of the hierarchy is inspired from the FilmFinder prototype [1]. Files having a type not defined in the legend are colored in gray. The coloration of each type of file can be deactivated and reactivated separately, by selecting the buttons of the legend. Selection and de-selection of one type of file in the legend make all the boxes of the files having the corresponding types blink.
- The last modification date of a file is an ordered variable that we visually coded using the brightness of the color from darker to lighter. The older files are darker.

We note that this usage of colors for visually displaying the characteristics of nodes would not be possible with the Cushion Treemap [5], another technique based on Treemaps.

Many options are provided for navigation and selection. PARENT visually highlights the path in the hierarchy when a node is selected using the mouse. It shows the depth of the selected element in the hierarchy. The path from the root to the corresponding node is displayed in black as shown in Figure 2: One file is selected at the bottom right of the display. We can observe the depth of the file in the hierarchy thanks to the displayed black stairs starting from the top title bar and going down to the selected file. If the selected element is a folder, the black stairs are displayed above and all the elements in the folder are highlighted in blue. In addition PARENT provides tools for the manipulation of files. As for the case of elastic windows [4], PARENT supports drag&drop actions for moving a file. Double clicking on a file will open it using the corresponding software in the Microsoft Windows environment. PARENT and Microsoft Windows Explorer are tightly coupled because a double click on a window title bar in PARENT launches Explorer and an option of the contextual menu of Explorer launches PARENT.

4 CONCLUSION

We have presented a new method to visualize hierarchical information. The method is based on standard nested Treemaps. Additionally the method enables the user to select the shape of the nodes bounding box of the tiled layout according to her/his task.

Usage of PARENT raises new challenges, including the selection of multiple files by defining a zone, the representation of symbolic links between files and the animation of the layout to visually show the reorganization after a modification of the hierarchy.

In future work, we plan to improve the evaluation function of our algorithm. One technique that we would like to experimentally test is to give different weights to each contained rectangles of an option. The notion of weight provides a simple mechanism for modeling importance. For example more importance can be given to the worst case so that an option containing a shape very far from the fixed one will be more likely eliminated. Another approach consists of defining the weight according to the domain characteristics of the elements. For instance the evaluation function can give more importance to files recently opened. As a result, the shapes of the boxes corresponding to the files recently opened will be closer to the fixed shape.

References

- C. Ahlberg, and B. Shneiderman, Visual Information Seeking: Tight Coupling Of Dynamic Query Filters With Starfield Displays. In Proceedings of *CHI'94* (Boston, MA, April 24-28, 1994), pages 313-317.
- [2] P. Fitts, The Information Capacity Of The Human Motor System In Controlling The Amplitude Of Movement. In Journal of Experimental Psychology 47, pages 381-391.
- [3] B. Johnson and B. Shneiderman, Treemaps: A Space-Filling Approach To The Visualization Of Hierarchical Information Structures. In Proceedings of Visualization'91 (San Diego, October, 1991), pages 284-291.
- [4] E. Kandogan E. and B. Shneiderman, Elastic Windows: Evaluation Of Multi-Window Operations. In Proceedings of CHI'97 (Atlanta, March 22-27 1997), pages 250-257.
- [5] J. van Wijk and H. van de Wetering, Cushion Treemaps: Visualization of Hierarchical Information. In proceedings of INFOVIS'99 (San Francisco, October 25-26, 1999), pages 73-78.



Figure 2: An extended Treemap of 1400 files is displayed in the PARENT system. The ratio is equal to 0.2: The corresponding horizontal rectangular shape is well suited to display a label. In the left part, three folders are displayed using an alternative view where the background of the three corresponding boxes is white. In the right part, a file called "Button2in" is selected using the mouse. The black stairs above the selected file highlight the four levels of depth of the file.



Figure 3: An extended Treemap of 240 files is displayed in the PARENT system. The ratio is equal to 1: The bounding boxes tend to be squares. In the left part, the files at the bottom are older than the files at the top: The boxes displayed at the bottom are consequently darker than the ones at the top. In the right part, a large folder named "Table" is selected. All the included elements are displayed in blue and the path from the root to the folder is highlighted in black.



Figure 4: The same hierarchy as in Figure 3 displayed using the standard nested Treemap algorithm of B. Johnson and B. Shneiderman.