

# Navigation Techniques for Zoomable Treemaps

Renaud Blanch and Éric Lecolinet  
ENST (GET) - CNRS LTCI UMR 5141  
46, rue Barrault  
F-75634 Paris cedex 13, France  
{blanch|elc}@enst.fr

## ABSTRACT

This paper presents a new technique called *zoomable treemaps* that makes it possible to navigate in very large trees. It integrates several efficient interaction techniques that enable multi-scale and structure-aware navigation.

**KEYWORDS:** multi-scale navigation, structure-aware navigation, zoomable treemaps, zoomable user interfaces.

## INTRODUCTION

The treemap visualization technique [5] is good at providing overviews of trees that have weights or sizes on the leaf nodes. But details are difficult to see when trees become large and deep, especially when the screen is small. With trees approaching the limit of a million items [2], the smallest details are not visible on a regular screen. For example, Figure 1 shows the hierarchical classification of web sites provided by the Open Directory Project (ODP) [1]. It consists of 694,986 categories distributed on 13 levels. The details of the *Arts/Music* category shown on Figure 1 (right) are not visible on the full tree shown on Figure 1 (left).

To solve this problem, we introduce the concept of *zoomable treemaps* (ZTM). Because of their recursive nature, treemaps can be seen as multi-scale representations that inherently define multiple levels of details. ZTMs enhance classical treemaps by using the zoomable user interface (ZUI) [3] paradigm to navigate efficiently in a hierarchical data space. Traditional ZUIs let users interact directly and continuously with the information space through panning and zooming. While this approach is quite natural, it requires much user interaction to allocate the maximum amount of available space to the nodes of interest. Many gestures are for instance needed to display efficiently the parent of a given node or one of its siblings with the “free flight” metaphor (by “efficiently” we mean that we want the new node to be entirely shown and to occupy most of the available space). We have designed a set of new multi-scale navigation techniques which make use of the specific structure of treemaps to facilitate navigation in such representations.

## NAVIGATION IN ZOOMABLE TREEMAPS

We have improved the traditional pan-and-zoom navigation techniques by using the structure of the treemap to guide the navigation and to make it “data-aware” [4]. We provide two kind of interaction techniques: discrete ones and continuous ones. Discrete interactions make it possible to select a particular node so that it will be displayed in the whole available

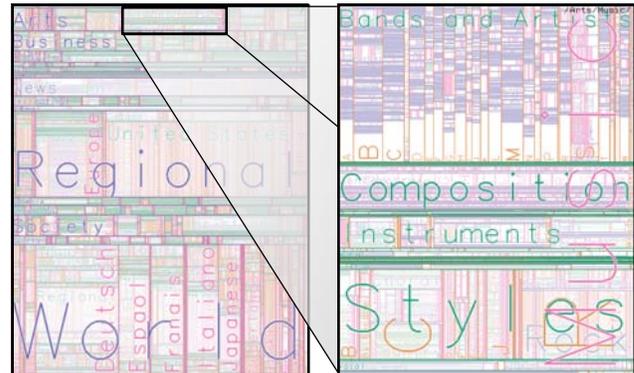


Figure 1: Tree displayed as a zoomable treemap.  
(left) 694,986 categories of the ODP [1],  
(right) zoomed view providing details on *Arts/Music*.

space. They trigger an animation that translates and scales the view continuously to fit the selected node, hence helping users keep their spatial orientation. Continuous interactions provide direct control on the view, and continuous feedback. Discrete interactions require less user input while continuous interactions provide more control. They both use the structure of the treemap to guide the navigation and thus enhance classical pan-and-zoom navigation techniques.

### Discrete Navigation: Node Selection

*Drilling-down/Rolling-up one level.* The discrete counterpart of zooming consists in *drilling-down* and *rolling-up* in the hierarchy. It is performed by left (resp. right) clicking to select a child (resp. the parent) of the current node (Figure 2).

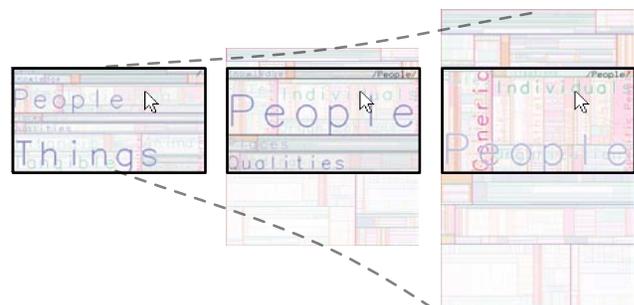


Figure 2: Drilling down one level.

*Flipping between sibling nodes.* The discrete counterpart of panning consists in *flipping* between nodes. The view is boxed in a frame that displays the neighborhood of the current node. Figure 3 shows three successive steps of the interaction: clicking (right) in a sibling of the current node displayed in the border frame “flips” the view (middle) to make

this neighbor become the current node (right). This interaction can be quickly repeated to browse through the children of a node as one would flip the pages of a book.



Figure 3: Flipping between sibling nodes.

*Through layer node selection.* Because several layers of the tree are displayed at the same time, not only the siblings and the children of a node are visible, but also all its descendants that have enough space to display their labels. *Through layer node selection* provides direct access to those visible nodes that would require a long pan-and zoom navigation to be accessed in traditional ZUIs. Hence *through layer node selection* speeds up navigation significantly by reducing the number of steps needed to access deep nodes.

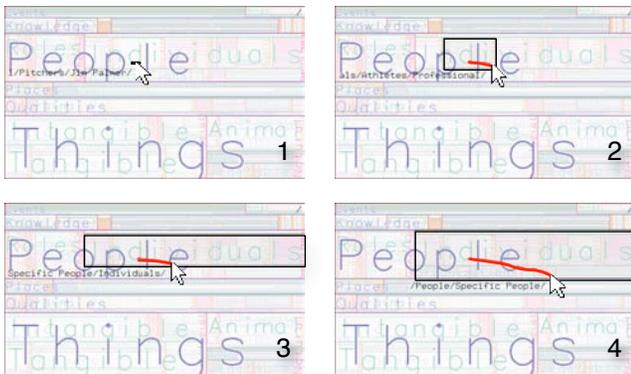


Figure 4: Through layers node selection using a crossing-based interaction.

Through layer node selection makes it possible to select any visible node by using crossing-based interaction. As long as users draw a stroke, the selected node is the smallest node that encloses the gesture. At the beginning, a leaf of the tree is selected (Figure 4.1), and each time the stroke crosses the bounds of the selection, its parent is selected in turn (Figure 4.2–4). At last, a stroke that ends in the frame surrounding the current node zooms out to the upper level.

#### Continuous Navigation: Enhanced Pan-and-Zoom

Continuous navigation complements the previous interaction techniques. It is based on traditional pan-and-zoom interaction, and thus allows a free manipulation of the view.

*Snap-zooming.* One difficulty with treemaps layout comes from the fact that the aspect ratio of the nodes and the one of the window that contains them can be very different. Pure geometric zooming is not well adapted in this case because the user may try to zoom in to a node that have a completely different shape from the window and thus will not fit well

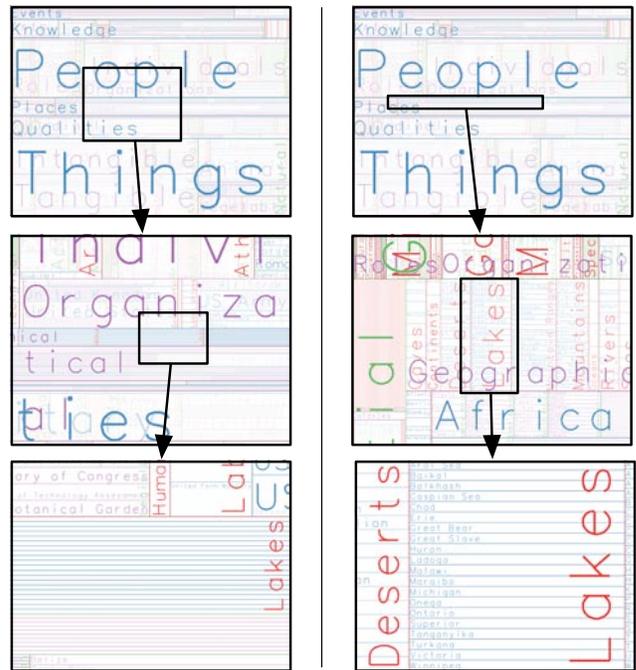


Figure 5: Snap-zooming. Steps of geometric zooming (left), and of snap-zooming (right).

inside (Figure 5 left). We propose a new technique, called *snap-zooming*, to solve this problem. Snap-zooming does not scale uniformly across  $x$  and  $y$  axis. Instead, it distributes the global scale factor that is controlled by the user in such a way that the aspect ratio of the target node becomes gradually similar to the one of the window (Figure 5 right).

#### CONCLUSION & FUTURE DIRECTIONS

Zoomable treemaps enhance traditional treemaps by making them multi-scale and thus suitable to explore very large trees. The interaction techniques that we propose facilitate navigation inside ZTMs because they use the structure of the tree to guide interaction. In future work we will evaluate our interaction techniques. We also want to adapt them to pen-based interaction on handheld devices.

#### ACKNOWLEDGMENTS

R. B. is funded by the Paris Île-de-France Regional Council.

#### REFERENCES

1. Open directory project. <http://www.dmoz.org>, Jan 2006.
2. J.-D. Fekete and C. Plaisant. Interactive information visualization of a million items. In *Proc. IEEE InfoVis'02*, pages 117–123, 2002.
3. K. Perlin and D. Fox. Pad: an alternative approach to the computer interface. In *Proc. ACM SIGGRAPH'93*, pages 57–64, 1993.
4. C. Plaisant, J. Grosjean, and B. B. Bederson. SpaceTree: Supporting exploration in large node link tree, design evolution and empirical evaluation. In *Proc. IEEE InfoVis'02*, pages 57–64, 2002.
5. B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Trans. Graph.*, 11(1):92–99, 1992.