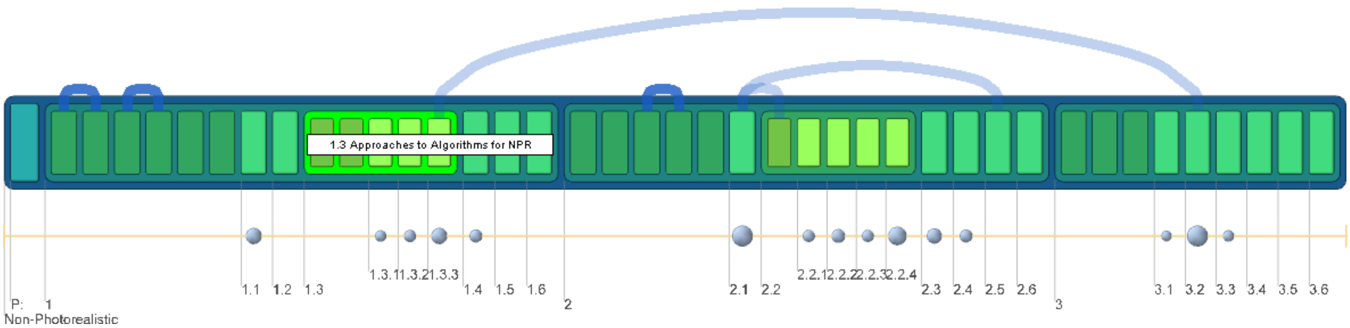
Interactive Poster: ArcTrees—Relations in Hierarchies

Petra Neumann*
University of Calgary

Stefan Schlechtweg†
University of Magdeburg

Sheelagh Carpendale*
University of Calgary



Abstract

Hierarchical data sets often include relations indicating natural or imposed dependencies between data items. Both types of relations, hierarchical and other, are often needed to understand the data. We present, ArcTrees, a novel way of visualizing hierarchical and non-hierarchical relations within one interactive visualization. An initial user study is described wherein ArcTrees compared favorably to the traditional Treemap visualization.

CR Categories: H.5.m [Information Interfaces and Presentation]: Miscellaneous—;

1 Introduction

Many types of data are either naturally hierarchical or can be grouped so as to establish a hierarchy. However, these relations are usually not the only types of relations of interest. While there are an increasing number of methods for drawing trees, almost no methods have been designed specifically to display the additional relations and reveal their connections to the hierarchy without resorting to general graph layout strategies. We present ArcTrees, a visualization method that integrates the display of additional relations with the hierarchical structure [4].

The development of ArcTrees was motivated by a need for understanding relations in structured documents. A structured document has an explicit hierarchical structure consisting, for example, of a book, its chapters, their sections etc. Additional relations might include forward and backward references or a suggested sequence of reading. Therefore, the general problem of this visualization is to combine two very different kinds of relational information in one visualization. The goal of such a visualization is to provide a visual tool that enables a better understanding of the structure of the data and that supports interactive exploration of the data.

Similarity relations between nodes are sometimes encoded through similar color, shape, or texture. Fekete et al. [1] overlaid a Treemap visualization with additional relations as Bézier arcs connecting two Treemap regions.

*e-mail: {pneumann | sheelagh}@cpsc.ucalgary.ca
†e-mail:stefans@isg.cs.uni-magdeburg.de

2 The ArcTree visualization

ArcTrees have been designed primarily as an informative interactive tool for augmenting information displays for digital media. Therefore, we set the following requirements for the visualization: use little screen space to leave most space for the media itself, reveal both the hierarchy and the additional relations, and include navigational tools and methods for exploring the visualization.

2.1 Visualizing the Hierarchy

We chose to represent the hierarchical structure via containment since this offers effective use of screen space. Leaf nodes in structured documents often have significant linear ordering, such as chapters in a book. Therefore, we combined the ideas of Treemaps [5] and Icicle Plots [3] to develop a "one-dimensional Treemap". An example of the layout and its metaphor can be found in Figure 1.

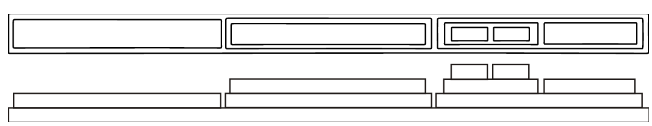


Figure 1: ArcTrees layout (top) and metaphor (bottom).

The display size of each node is influenced by a given node metric. Node metrics can space nodes according to structural information like the number of descendants of a node or depend on the data contents of the node or some value provided via interaction (degree of interest). Node sizing can be interactively adjusted. Color coding is used to portray structural information and for highlighting.

2.2 Visualizing Relations

Relations are given as pairs of nodes. Inspired by Arc Diagrams [6], for each relation we draw an arc between the horizontal centers of the two nodes. Curves can be controlled in height to allow for flexible usage of display space. Additional information can be encoded using the opacity and width of the arcs. All arcs are drawn somewhat transparent to make the crossings clear and understandable. A decreased opacity encodes connections between collapsed nodes

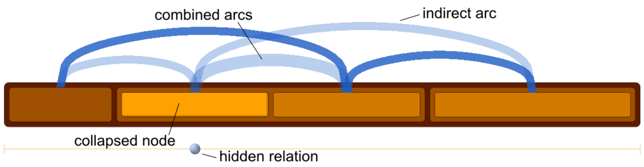


Figure 2: Encoding techniques used for relations

3 INTERACTION

To enable the exploration of large hierarchical data structures, interaction techniques need to augment the spatial layout. Appropriate interaction tools are critical because the number of leaf nodes that can be displayed is bounded by the display resolution. As the number of nodes gets larger visual clutter increases and individual nodes may get too small to be discernible. To address this problem we provide a variety of interaction techniques including Zoom+Filter and Focus+Context techniques for both the hierarchy and the relations.

We implemented a simple Zoom+Filter mechanism by expanding and collapsing subtrees based on user interaction. A collapsed subtree is represented as a button with shading to indicate that it can be pushed. A fully expanded or leaf node is drawn as flat (Figure 2).

Subtrees may also be expanded or collapsed according to their assigned degree of interest (DOI). The DOIs for each node are computed based on Furnas [2] with an adjustment to account for the connection of nodes with additional relations. Relations between nodes contribute to the DOI value so that nodes connected to the focus receive a higher degree of interest than unconnected nodes. The DOI values can also be used as a node metric (cf. Section 2.1), thus, yielding a visual Focus+Context technique. When interacting with the tree visualization, nodes that are connected via relations may become invisible or visible. To avoid information loss connections to hidden nodes also need to be shown. When a node connected by a relation becomes hidden during interaction, its connecting arc is drawn to the collapsed parent. If at least one of the nodes directly connected by a relation is hidden inside a collapsed node, the arc representing this relation is drawn using a lower degree of opacity (Figure 2). The case in which a relation is completely hidden inside a collapsed subtree is indicated by a circular glyph below the collapsed node to indicate that hidden relations exist (Figure 2).

4 INITIAL EVALUATION

The space usage of ARCTREES has been limited to a narrow strip approximately one tenth the size of a normal desktop display. This raised many concerns according to the readability of the layout. In a preliminary user study we, therefore, tested if the ARCTREE layout could improve the depiction of tree topology compared to the traditional Treemap layout. The study aimed at finding differences in performance of both layouts in depicting global and local structural information of tree data. The study was conducted with ten participants using a fully-crossed within-subjects 3 (tree size) x 2 (layout) design. We hypothesized that ARCTREES would perform better according to answer time and accuracy in typical real world tasks involving finding certain nodes in a tree, relating them to their context (ancestors, siblings, and descendants), or finding groups of nodes by identifying patterns in the tree structures.

An ANOVA ($\alpha < 0.05$) found significant differences in mean answer time between the two layouts (Figure 3) while answers were given with an approximately equal error rate. This suggests that even though the ARCTREE display uses less display space, it performs better or equal in comparison to the Treemap layout. Certainly, this study is only a first step in a more careful evaluation of the ARCTREE visualization. Future studies should include other tree visualizations, tasks, or tree data.

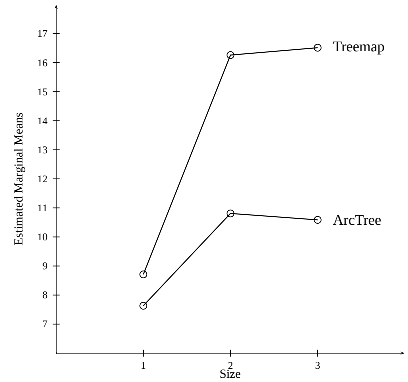


Figure 3: Estimated marginal means (response time) for layout*size.

We envision ARCTREES to be used as a modular component of an electronic reading environment. With the given the space constraints, it is imaginable that such a visualization might be part of, e. g., Acrobat Reader as the thumbnail overview is today (Figure4).

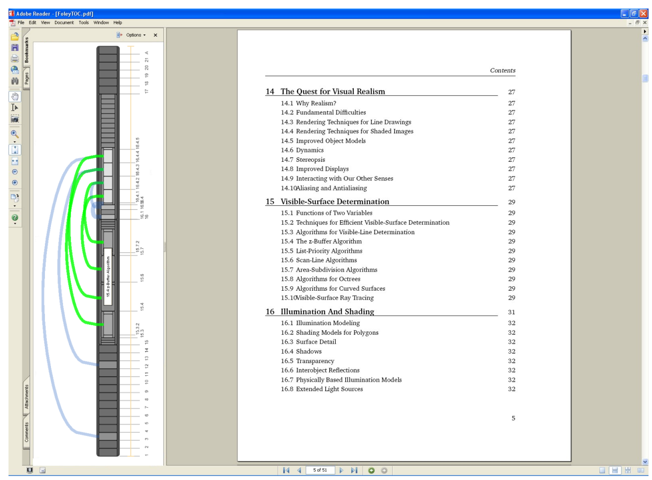


Figure 4: A possible embedding in a reading environments.

5 ACKNOWLEDGMENTS

This work is supported by NSERC, iCORE, and Alberta Ingenuity.

REFERENCES

[1] Jean-Daniel Fekete, David Wang, Niem Dang, Aleks Aris, and Catherine Plaisant. Overlaying Graph Links on Treemaps. In *Proc. InfoVis'03, Poster Compendium*, pages 82–83. IEEE Press, August 2003.

[2] George W. Furnas. Generalized Fisheye Views. In *Proceedings of CHI'86*, pages 16–23, New York, 1986. ACM SIGCHI.

[3] Joseph B. Kruskal and James M. Landwehr. Icicle Plots: Better Displays for Hierarchical Clustering. *The American Statistician*, 37(2):162–168, May 1983.

[4] Petra Neumann, Stefan Schlechtweg, and M. Sheelagh T. Carpendale. ArcTrees: Visualizing Relations in Hierarchical Data. In *EuroVis 2005*, pages 53–60, 319. Eurographics, 2005.

[5] Ben Shneiderman. Tree Visualization With Treemaps: a 2-d Space-Filling Approach. *ACM Transactions on Graphics*, 11(1):92–99, 1992.

[6] Martin Wattenberg. Arc Diagrams: Visualizing Structure in Strings. In *Proc. InfoVis'02*, pages 110–116. IEEE Press, 2002.