Symmetry and Node Focused Visualization of Large Trees

Katayoon Etemad*
University of Calgary

Sheelagh Carpendale†
University of Calgary

ABSTRACT

In this paper, we take a different approach to visualizing very large trees. To facilitate presentation and exploration of massive hierarchical datasets such as linguistic and genealogical hierarchies, our approach considers drawing layouts of *tree-cuts* as a function of a node-of-interest or NOI, and uses interaction to support rapid access to the entire tree. Instead of emphasizing overall tree structure, our layout is designed to make the most space available for the node-of-interest and its immediate ancestors and descendants. Inspired from Persian floral patterns, we describe the development of *ShamsehTree* and *PaisleyTree*, showing how the use of symmetry can provide new structures for tree layouts.

Keywords: Information visualization, trees, symmetry, graphs.

Index Terms: H.5.2 [Information Interfaces and Presentation]: User Interfaces—Graphical user interfaces (GUI) I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

1 INTRODUCTION

Creating readable tree layouts for large datasets has long been a challenge. Far from reaching resolution, this challenge continues to intensify as datasets continue to expand exponentially, and, while display sizes are increasing, the discrepancy between the available display space and the amount of data to be displayed also continues to grow. As a result, visualization of large hierarchical datasets continues to be an active research area. Visualizations of hierarchical data usually focus on conveying structure. However, with really large hierarchies, such as WordNet [7], Mathematical Genealogy [9], Citation Patterns [2] and Evolutionary Trees [12] laying out the structure can require more space than there are pixels in the display. In contrast, we consider the idea of node-focused layouts [5] combining tree-cuts with symmetric floral patterns to create a new interactive node-focused layouts.

The concepts that influence our creation of new layouts include drawing from and combining these disparate ideas: expanding the use of symmetry in the layout, developing layout structures based on traditional aesthetic patterns, creating new types of interactive tree-cuts, and combining nested, node link and adjacency tree layouts to create Node Focused Tree-Cuts, *ShamsehTree* and *PaisleyTree*.

Symmetry is a pervasive phenomenon in nature and art. It has long been recognized as playing an integral role in geometry and architecture as well as being an important factor for harmony, balance, and proportion. A whole range of symmetries exist such as translational, rotational, reflectional and dilational symmetries. Symmetry has been studied in human perception [13] and has been discussed as an important factor for recognizing, and grouping elements within a representation [11]. There has been some exploration of the use of symmetry in graph drawing [11, 4], and the impact of translational symmetry on human vision system has been formulated as one of the Gestalt principles [15]. However, the potential and power of symmetry has not yet been fully explored.

The idea of incorporating aesthetics in visualizations has been a persistent stream of research in information visualization. Aesthetically appealing representations of information has been shown to better engage and motivate the viewer [16]. For example, a Mondriaan style layout has been used to represent family members emails [8]. *InfoCanvas* [10] offers people the opportunity to create their own information canvas by adding information representation for factors such as stock market values and weather information to a landscape scene. In our tree layouts we have been inspired by Persian floral patterns (Figure 1) and have approached using them to generate new layouts by first algorithmically deconstructing a traditional aesthetic pattern. We then use this deconstruction to create building blocks for new layout approaches.

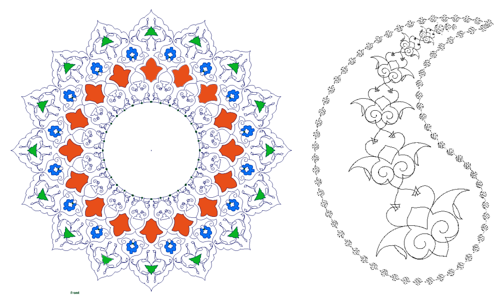

Figure 1: *Left: Concentric floral patterns, known as Shamseh in Persian floral patterns [6]. Right: Dilational symmetry in Persian floral pattern known as Boteh or Paisley.*

To motivate the use of tree-cuts, consider that when the size of the tree exceeds the display resolution, visualization choices are necessary. Two common strategies are; one to display only a part of the tree with enough detail (local context), or two to display the entire tree without all the detail (global context). Both strategies lead to questions about selecting which part of the tree to display and what interaction methods would be suitable. Important interaction problems include the navigation between global context and local detail, and the interaction required to change the focus. To address the challenge of data size, we suggest a tree-cut method that creates a node-focused sub-tree that presents only specific parts of the tree in proximity with the NOI.

Traditionally tree layouts use one of three types of graphic representations to express relationship among the tree nodes. These are the use of links or edges between nodes, the use of nesting or implying a relationship by placing one node inside another, or the use of adjacency in which relationships between nodes are declared by placing related nodes so that they touch. In the design of the ShameshTree, we use a nested layout based on contracting circles in conjunction. The NOI is presented in a circle and the decendant subtree rooted at NOI is presented inside the NOI's circle. The ancestor nodes of NOI are presented with concentric circles in such a way that each parent's cilcle is slightly bigger than its own child's circle. The sibling nodes of the NOI are also presented with a similar structure as NOI, in circular arrangement around the NOI and inside their parent's circle.

*e-mail: ketemad@ucalgary.ca
†e-mail:sheelagh@ucalgary.ca

In the design of PaisleyTree, we use adjasency and node link in addition to nested layout for presenting the tree-cut. In this implementation the NOI is presented in a circle with two levels of its desendent subtree. A specific level of NOI's ancestor nodes are presented with contracting circles. The more distant the ancestor is from the NOI, the smaller its circle become in the layout.

2 Designing Node Focused Tree-Cut Layouts

In this section, we explain the design decisions that led to our Node Focused Tree-Cut Layouts. Both *ShamsehTree* and *PaisleyTree* are Node Focused Tree-Cut layouts and were based upon these design decisions.

2.1 Addressing Data Size

A Node Focused Tree-Cut layout is based on two concepts. First, it may be important to see the details of a particular node, or NOI. Second, it may be useful to display only a portion of the whole tree, a tree-cut. In extremely large trees, representing overall structure can make access to and discovery of individual nodes difficult, due to limited available space. To address this, in our Node Focused Tree-Cuts layouts we emphasize a NOI by making more space available for the chosen node and develop a tree-cut by displaying the hierarchical structure of the tree around the NOI. This is a new kind of tree-cut that presents a subset of the tree in the proximity of the NOI. In this approach, the ancestor and descendant nodes are visible depending on their distance from the NOI.

Specifically, the tree-cut in ShamsehTree implementation includes all the direct ancestors from the NOI to the root of the tree, plus two levels of their descendants. Similarly, two levels of the NOI's direct descendants are displayed. The tree-cut in PaisleyTree implementation on the other hand includes a specific number of ancestor nodes of NOI plus two levels of their descendants.

This tree-cut strategy reduces the amount of clutter on the screen. The conventional tree-cut techniques [1, 3] consider only a subtree of the hierarchy starting with a particular node as the root of the subtree. In this kind of representation, the hierarchy above the root of the subtree is not presented. Consequently, the hierarchal relationship between the node at the root of the subtree and its ancestors is missed.

2.2 Addressing Edge Clutter

In large tree visualization edges are one of the sources of cluttering. Too many edges can cross each other and cover other nodes. Considering this problem our intention is to not represent edges explicitly, but show them implicitly. The primary approach to addressing edge clutter in *ShamsehTree* is to avoid the need to draw edges by using a nested layout. While in the PaisleyTree, adjacency and some minimal use of node links are used.

2.3 Interactive Navigation

Navigation in a tree is always an important factor; however, this is especially true when the tree is represented by a tree-cut. It must be possible to reach other parts of the tree readily. The choice of tree-cut is an important factor in this. The fact the all direct ancestors of the NOI are shown plus two levels of their children makes it possible to move both higher in the tree and latterly very rapidly. Need the Paisely statement here

3 ShamsehTree

In the ShamsehTree tree-cut, if node A is inside node B then A is a child of B. This method is known as nested layout for tree visualizations. Instead of representing a whole tree, *ShamsehTree* represents the partition of a tree that is close to the viewer's NOI. Figure 2 illustrates this layout in comparison with a traditional, node-link, tree layout.

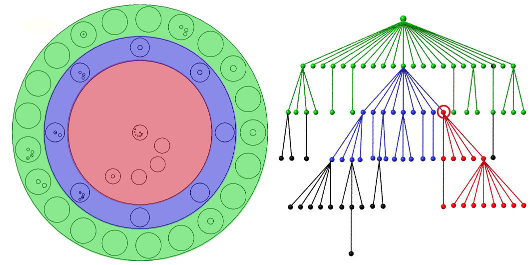

Figure 2: *Schematic comparison between the* ShamsehTree *and a traditional node link tree layout.*

In Figure 2, for both the node link and the nested layouts, the section in red represents the NOI and its children. The blue section represents the NOI's parent and its children. All of the NOI's siblings are located around the NOI and inside their parent's node. The section that is shown in green is the NOI's grandparent which is the *root* in this example, including all of its children for two generations. As it is demonstrated in the figure, in this representation, the whole tree may not be visible at the time, but viewers can explore and move to their desired nodes and its related descendants and ancestors. *ShamsehTree* layout is based on concentric circles starting from the root as the outer circle down to the NOI as the inner circle. If the NOI has children, they are laid out inside the NOI using Phyllotactic patterns. If the NOI has siblings, they are evenly spaced in the NOI's parent's ring using rotational symmetry. This is repeated for all ancestor nodes of the NOI. The NOI and all drawn ancestor nodes are displayed with specific levels of their descendant subtree. In the current implementation, two levels of descendants are visible. For the NOI, this creates both an ancestrally-based and a descendant-based context.

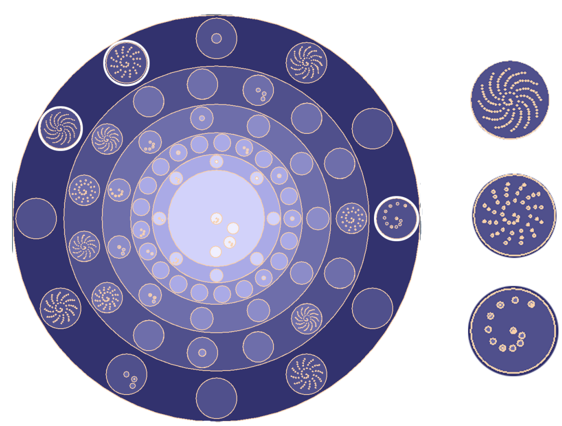

Figure 3: *The three nodes shown on the right, from top to bottom, have 156, 50, and 11 children, all of which have bee, laid out in the same size parent. For all three nodes α is the same. Note how for the node with 11 children, only the inner part of the spiral is used.*

When a node A is selected, that node becomes the NOI and moves dynamically to the center of the layout and its size is appropriately increased. Following this action, the NOI's children are recursively resized and re-positioned. Since the NOI's siblings are to be arranged within the NOI's parent node B, the size of the NOI must be slightly smaller than its parent to leave enough space for the siblings. We uniformly space the siblings of A in the space around A and inside B as demonstrated in Figure 3 (left side). The NOI and its siblings are displayed with the same color.

And the size of the NOI is increased by $r_A = d\,r_B,\ 0 \le d \le 1$, where d is a constant that controls how much smaller the NOI is than its parent. As illustrated in Figure 4, larger values for d (left) provide more space for the NOI and smaller values for d (right) provide more space for ancestors and their children (NOI's siblings). However, really deep trees large values of d do not provide enough space for the NOI.

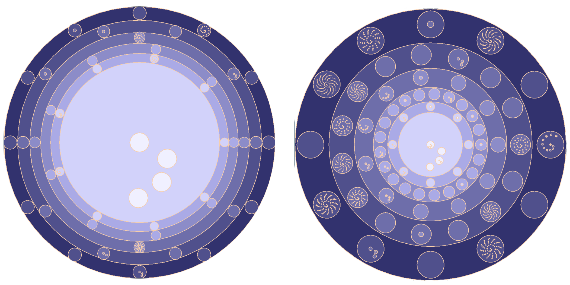

Figure 4: *Left: larger values for d gives more room for an NOI to be deeply nested in the tree; Right: smaller amounts for d provides more space for the ancestors and their children.*

In addition, extremely small or large values for d do not produce proportional and aesthetically appealing layouts. Setting d to the values near to the golden ratio (the right layout in Figure 4) seems to make more aesthetically pleasing layouts.

To explain the positioning of the siblings, assume $S_1, S_2, \ldots, S_k$ denote siblings of NOI. As a simple solution, we distribute these nodes on a circle uniformly around the NOI and within their parent. More precisely, we use the new NOI's parent's center (x_B, y_B) as the new NOI center and to obtain the position of the sibling's centers we calculate a sibling-base circle upon which all siblings will lie.

4 PaisleyTree

The PaisleyTree tree-cut is similar to the ShamsehTree tree-cut. It is different in that it limits the number of ancestors shown in the layout of the NOI to six, instead of portraying all of the ancestors until the root is reached. This ensures that the limited space of the layout focuses on the NOI without becoming crowded like the ShamsehTree layout. The dilation and shrinking as well as the Paisley pattern's shape imply the presence of the ancestor nodes that have been cut but they are not actually presented. The apex of the Paisley shapes implies the presence of the root node. This implied presence can be used for interaction.

Figure 5 shows three diminishing nodes of a PaisleyTree on the left and a more traditional node-link layout of the tree on the right. The right-hand node-link tree drawings use color coordination to highlight the nodes that are represented in the three PaisleyTree nodes on the left-hand side. Figure 6 also shows a comparison of a node-link layout on the right-hand side with the PaisleyTree on the left. In the PaisleyTree the NOI is colored red and its ancestors are blue. In the node-link drawing the colors correspond, highlighting the portion of the tree included by the PaisleyTree tree-cut.

4.1 Hybrid Layout

In this section, we describe our use of adjacency, nesting, and node-link techniques to create a hybrid layout.

4.1.1 Adjacency Layout

For the NOI, we present its ancestors as a sequence of shrinking circles whose centers are positioned on the spiral of the Paisley, from C to A. Therefore, the farther the ancestors are from the NOI the smaller and closer to the apex they become. Since we would like to use the area of the circles to present descendants of these nodes,

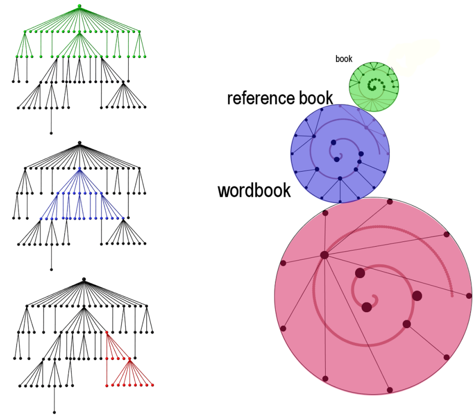

Figure 5: *Comparing the traditional tree layout with PaisleyTree, data from the "Book" subtree of the* WordNet *[7] .*

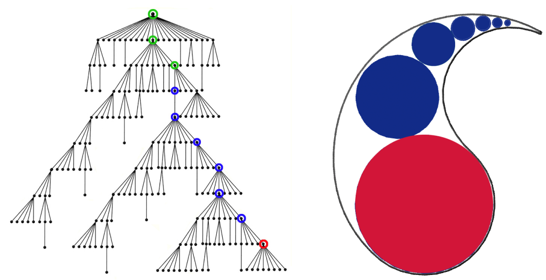

Figure 6: *Comparing the display of ancestors of a traditional tree layout with that of a PaisleyTree. The NOI is shown in red.*

the circles should not intersect one another. In addition, to optimize the use of the space, we enforce consecutive circles to be tangent to each other. This arrangement of circles is an adjacency layout. Figure 6 compares a traditional node-link tree with this shrinking circle layout. Note that for both diagrams the NOI is shown as red, the NOI's ancestors are shown in blue when there is corresponding space in the Paisley layout, and are shown in black where they have been pruned from the Paisley layout. This is true for the ancestor node's descendant structures as well.

4.1.2 Nested Internal Node Layouts

Thus far we have described the layout structure for the NOI and its immediate ancestors. These nodes have been laid out adjacently within the paisley pattern's boundary, diminishing as they recede from the NOI according to dilational symmetry. The descendants are laid out by using nested and node-link layouts within these adjacent nodes. For each of the NOI and its adjacent ancestors two generations of descendants are presented.

For nested internal layouts that distribute subtrees inside of adjacent circle nodes, we propose a hybrid design that is a combination of nested and node link layout. The subtree of each node is positioned inside the circle of that node while the relationship between two levels of its children is presented with explicitly use of edges.

4.1.3 Node-Link Layout

In our node-link internal layout children and grandchildren are positioned inside of the node's circle. The grandchildren are positioned around the perimeter of the circle. The relationships between the children and grandchildren are represented by explicit edges as in a node-link layout.

The size of the descendant nodes is determined separately for the children and grandchildren as a function of the number of nodes in each category. A large number of nodes are drawn smaller than a small number of nodes.

To better address this arrangement of internal node-link nodes we developed another node-link layout where the children are positioned along a spiral within the circle (as shown in Figure **??**). Grandchildren are still placed along the circle's perimeter. Positioning on a spiral shape helps present children across the node's entire area, especially when presenting large descendant trees.

To avoid overlapping edges, leaf children (children that do not have any children of their own) are positioned on the inside of the spiral while the rest of the children are positioned along the outer loop of the spiral. Each child is connected to all of its children with straight lines (Figure **??**). Explicitly presenting the spiral base helps indicate that all children on the spiral have the same relationship with the circle's node.

To create this layout we begin by calculating the number of children and grandchildren the selected node has. Based on the number of grandchildren, we define an angular increment for positioning them along the edge of the circle. Only nodes without children (i.e,. leaves) are placed inside the spiral.

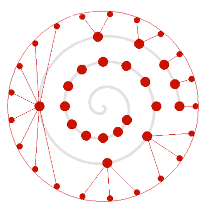

Figure 7: *A schematic representation of placement of the children and grandchildren of the NOI in the PaisleyTree layout.*

4.1.4 Bringing It Together: A Hybrid Layout

As we now consider the PaisleyTree layout as a whole we can see that we have used an adjacency layout for presenting NOI and its visible ancestors. Inside these adjacent nodes we make use of nested node-link or nested layouts to present the subtree of each adjacent node and its descendants.

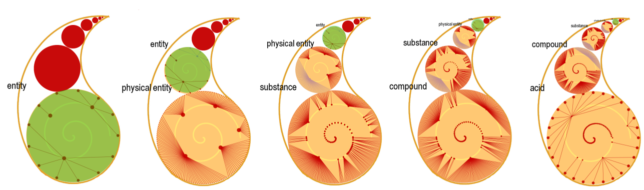

Figure 8: *Interaction with PaisleyTree. Five steps of interaction, changing the NOI from the root (left) to its fifth generation descendant (right). The root node has been colored green to clarify the progression between nodes-of-interest.*

Figure 7 illustrates the interaction with PaisleyTree in several steps. The tree used in the Figure is the *entity* subtree of the *WordNet* [7] tree that features 75111 nodes and 18 levels. In the leftmost image the root node, *entity*, is the NOI.

5 Future Work

There are several possible directions for future work. Perhaps the most important is to investigate to what extent these layouts might be useful in practice. Studying several aspects such as ease of navigation, and ease of search would be interesting. We are particularly interested in the effect that the use of different symmetries and different colors might have on navigation and search.

For the layouts introduced in this project, many Persian floral patterns have been examined and finally Shamseh and Paisley patterns were selected. However, other patterns have potential for being selected as a design for tree layouts. Moving beyond these patterns by exploring western floral patterns and Islamic star patterns would be another possible direction for future work.

One can also consider large graph visualizations. Since graphs do not necessarily have hierarchical structure, the nested layout may not be usable for graph layouts. However, the general ideas of the use of symmetries and node-focused layouts, can be explored as possibilities for visualizing large graphs.

The role of color in these layouts is another interesting direction to investigate. Color offers many possibilities for varying emphasis and attention [15]. Designing layouts with different uses of color and follow through with some empirical studies with the goal of understanding these different uses is another possibility for further research.

References

[1] A. Barron, J. Rissanen, and B. Yu. The minimum description length principle in coding and modeling. *IEEE Transactions on Information Theory*, 44(6):2743–2760, Oct 1998.

[2] C. Bergstrom and M. Stefaner. Well-formed. eigenfactor.org. http://well-formed.eigenfactor.org/index.html, September 2009.

[3] Y. Cao, H. Duan, C. Lin, Y. Yu, and H. Hon. Recommending questions using the mdl-based tree cut model. In *Proceeding of the 17th International Conference on World Wide Web, (WWW'08)*, pages 81–90, NY, USA, 2008. ACM Press.

[4] P. Eades and K. Sugiyama. How to draw a directed graph. *Journal of information processing*, 13(4):424–437, 1991.

[5] K. Etemad and S. Carpendale. Shamsehtrees: Providing hierarchical context for nodes of interest. In *Proceedings of Bridges 2009: Mathematics, Music, Art, Architecture, and Culture*, pages 293–300, 2009.

[6] K. Etemad, F. Samavati, and P. Prusinkiewicz. Animating persian floral patterns. In *Proceedings of Computational Aesthetics in Graphics, Visualization, and Imaging, (CAe'08).*, pages 25–32, 2008.

[7] C. Fellbaum. *WordNet: An Electronic Lexical Database.* MIT Press, Cambridge, MA, USA, 1998.

[8] L. Holmquist and T. Skog. Informative art: information visualization in everyday environments. In *Proceedings of the 1st International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, pages 229–235, New York, NY, USA, 2003. ACM Press.

[9] M. Keller. Mathematics genealogy project. http://genealogy.math.ndsu.nodak.edu/index.php, September 2009.

[10] T. Miller and J. Stasko. Artistically conveying peripheral information with the infocanvas. In *Proceedings of the Working Conference on Advanced Visual Interfaces, (AVI'02)*, pages 43–50, New York, NY, USA, 2002. ACM Press.

[11] P. Neumann, S. Carpendale, and A. Agarawala. Phyllotrees: Phyllotactic patterns for tree layout. In *Proceedings of Eurographics / IEEE VGTC Symposium on Visualization, (EUROVIS'06)*, pages 59–66. Eurographics, 2006.

[12] A. Phan, B. Tseng, E. Suh, and K. Tseng. Bench press. http://blog.benchside.com/?s=Tree+2.0, September 2009.

[13] J. Thorpe and P. Oorschot. Graphical dictionaries and the memorable space of graphical passwords. In *Proceedings of the 13th USENIX Security Symposium*, pages 10–17, Berkeley, CA, USA, 2004. USENIX Association.

[14] W. Wang, H. Wang, and G. Dai. Visualization of large hierarchical data by circle packing. In *Proceedings of the Conference on Human Factors in Computing Systems, (CHI'06)*, pages 517–520, New York, NY, USA, 2006. ACM Press.

[15] C. Ware. *Information Visualization: Perception for Design.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.

[16] K. Zhang. From abstract painting to information visualization. *Computer Graphics and Applications*, 2007.