

Tagging & Hierarchy | Studio Task #1 (Individual)

Due: Thursday, April 4, 2013

Goals

The goal of this assignment is to build a novel visualization for viewing/accessing hierarchical or tagged data.

Assignment

Hierarchy is fundamental to computing systems. It is the basis for abstraction, information organization, and software/hardware design. Hierarchy also manifests itself in life—observable in government, business, language, etc. Nearly all information can be arranged in some sort of hierarchical structure. According to the Universal Principles of Design book, “hierarchical organization is the simplest structure for visualizing and understanding complexity” (p. 104). Thus, one thrust in information visualization design has been to create interfaces/metaphors to arrange and present hierarchical data.

The Web 2.0 phenomenon of tagging (e.g., Flickr tags or gmail labels) has been seen as an alternative to hierarchical representation. For this design homework, we’d like you to think, first, of some data that you interact with everyday (e.g., your computer’s file system, e-mail, or web links). **Then, come up with a novel design of how this data could be visualized and navigated either using tags or hierarchy (or both).**

The design may be paper based or mocked up in PowerPoint, Photoshop, or some other graphical tool. If you choose to design your interface on paper, please scan in your design so that we have a digital artifact. Your designs should be made accessible via your individual studio project web site (see below). You will be presenting your design in class. The presentation should include sketches of your interface and justifications for your design.

You might want to consider what makes certain data types amenable to tagging versus hierarchy. For example, would a tagged file system be superior to a hierarchical one? Are tagging and hierarchy mutually exclusive? Would a hybrid system be possible? If you choose to build a hierarchical system, be prepared to talk about whether or not a similar design would work for tagging (or vice versa). Please read the [2-page hierarchy reading before you begin](#).

I’ve included some example designs for organizing / visualizing information. Feel free to incorporate aspects of these systems or to extend an existing interface that you find on the web into your designs.

Hierarchical Visualizations

The book refers to three basic ways to visually represent hierarchy: trees, nests, and stairs. Below, I include some relatively novel examples of these techniques; not all of which are clearly advantageous over their more pedestrian counterparts.

A Tree: Cone Trees

Link: <http://portal.acm.org/citation.cfm?id=108883>

Description: Cone Trees visualize hierarchical information in 3D to “maximize effective use of available screen space and enable visualization of the whole structure.” The visualization uses animation to shift some of the user’s cognitive load to the human perceptual system. The lucidity of such a design is in debate, however. The Universal Principles of Design book, for example, argues that “representing hierarchical structures in three-dimensional space improves little in terms of clarity and comprehensibility—though it does result in some fascinating structures to view and navigate.”

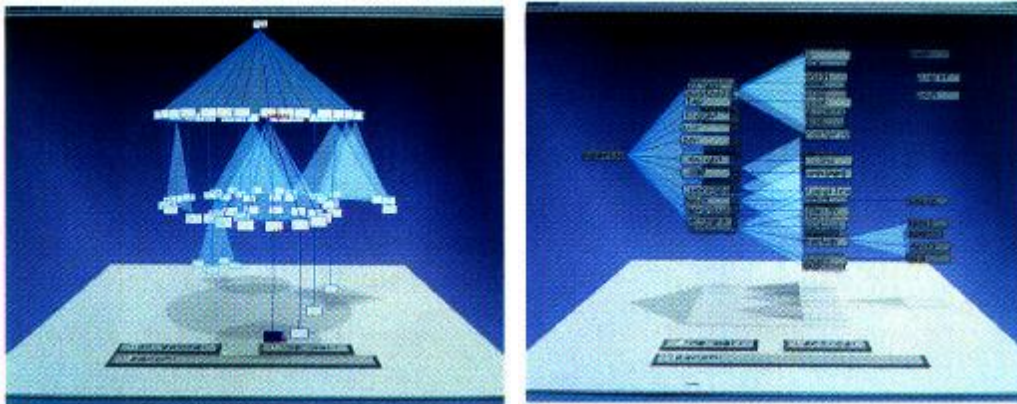


Figure 1: Cone Tree courtesy of [Cone Trees: Animated 3D Visualizations of Hierarchical Information](#).

A Tree: Prefuse's TreeView:

Link: <http://www.prefuse.org/gallery/treeview/>

Description: Like Robertson's Cone Tree, Prefuse's TreeView uses animation to reduce cognitive load and guide the user in navigation. The link above points to a Java demo of the interface. Both Windows and the Mac OS X have incorporated animation to help structure interaction and improve interface aesthetic.

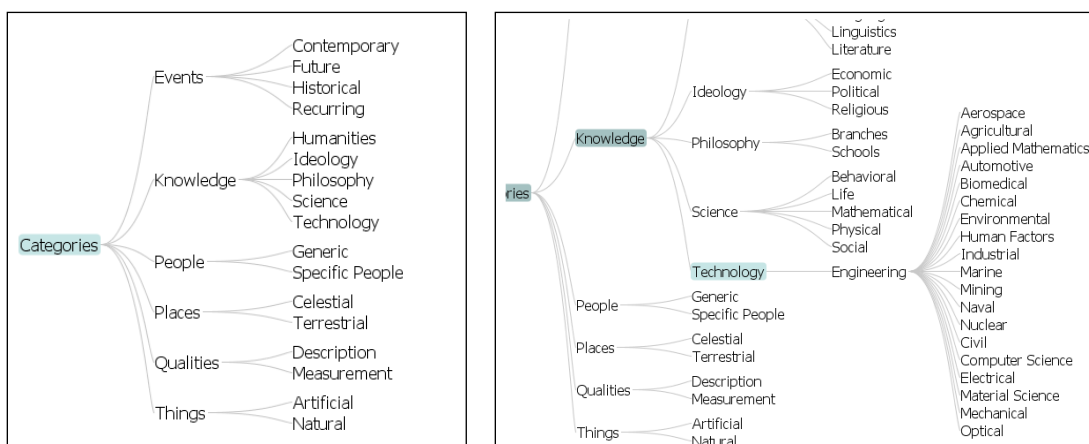


Figure 2: Prefuse's TreeView courtesy of <http://www.prefuse.org/gallery/treeview/>

A Nest: TreeMaps

Link: <http://www.cs.umd.edu/hcil/treemap-history/index.shtml>

Description: The term “TreeMap” refers to the notion of turning a tree into a planar space-filling map. TreeMap uses a recursive algorithm for layouts, where the spatial footprint is representative of the visualized entities size. Color is also used to provide an additional feature dimension. According to Shneiderman, TreeMaps’ inventor, the average user takes about 10-15 minutes to become acquainted with the system.

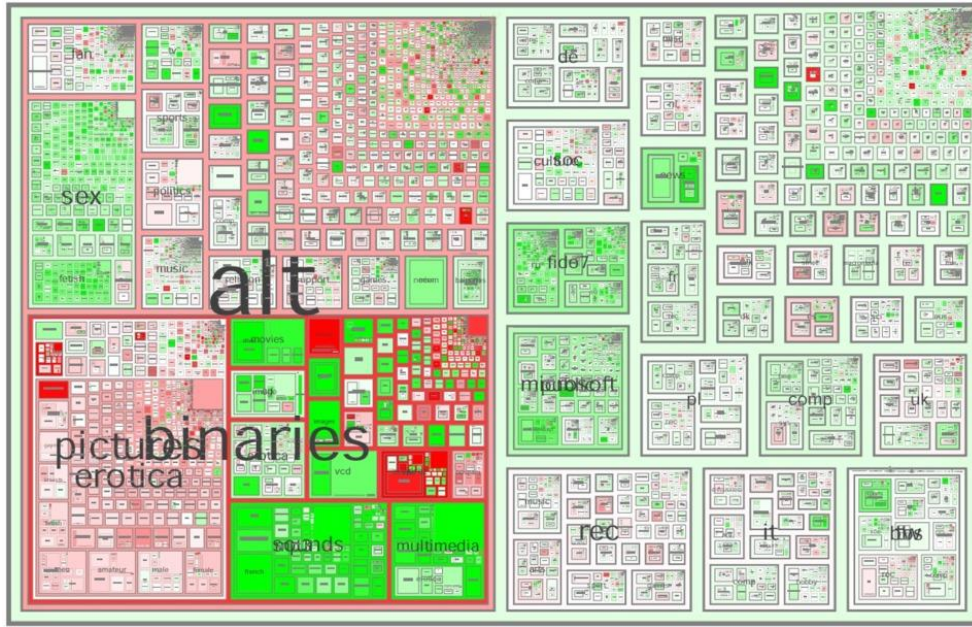


Figure 3: Tree Map courtesy of [Visualization Components for Persistent Conversations](#)

A Stair: Windows Explorer

Description: See reading.

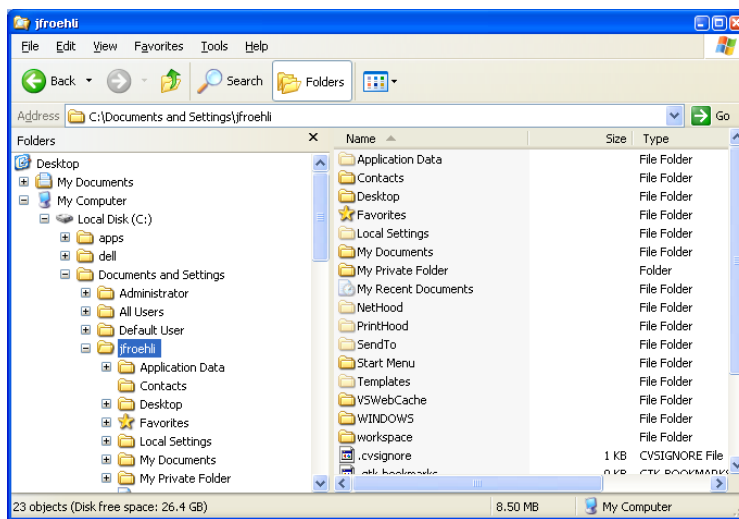


Figure 4: A Stair visualization

Tagging Visualizations

Tagging has become intricately tied to the Web 2.0 phenomena. Tags themselves are not necessarily novel (their roots are in “labeling” and “attributes” meta-data fields); however, they have forced designers to rethink the visual representation of data.

Tag Cloud

Link: <http://www.flickr.com/photos/tags/>

Description: A tag cloud is a weighted list of words, where, typically the most frequently used tags are depicted in a larger font or otherwise emphasized. The display order tends to be in alphabetical order. Note, however, that there is no hierarchical arrangement to a tag cloud—though one could imagine applying hierarchy (e.g., some hierarchical grouping).

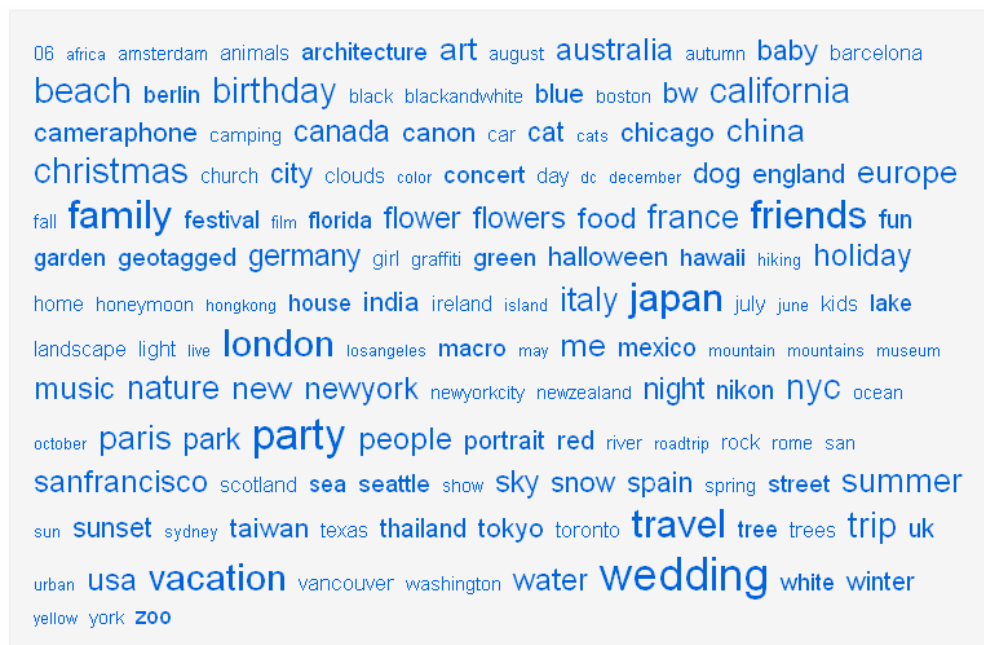


Figure 5: The Flickr tag cloud system, link: <http://www.flickr.com/photos/tags/>

Novel Representations...

The example(s) below don't necessarily use tagging or hierarchy to present data.

BumpTop:

Link: <http://en.wikipedia.org/wiki/BumpTop>

Description: BumpTop is a prototype interface for a computer's file system, which behaves in a more realistic manner by adding physics simulation and piling instead of filing as the fundamental organizational structure. Objects can be casually dragged and tossed around, influenced by physical characteristics such as friction and mass, much like we would manipulate lightweight objects in the real world.

Is this metaphor useful? At what point should a visualization / interaction metaphor distance itself from reality to leverage the power of a computer. What are the limitations of the design below?

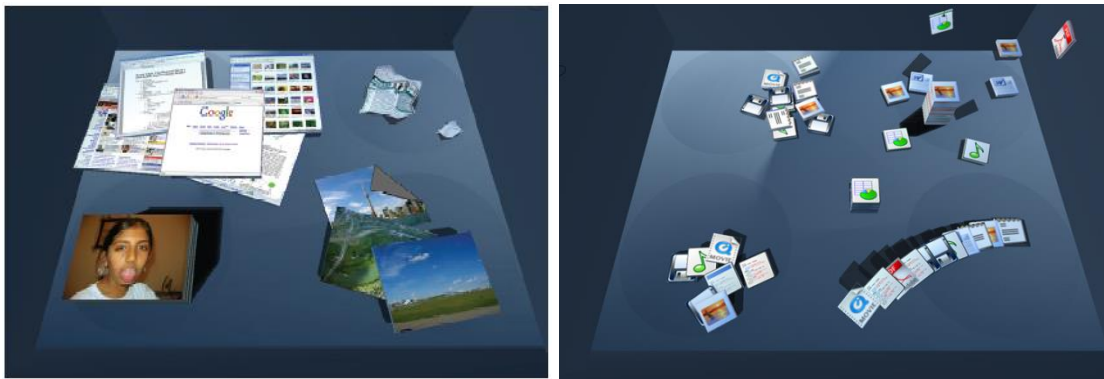


Figure 6: The BumpTop interface

Deliverables

1. Create your studio design website

Each student has been assigned an individual project directory. You can access your directory either via ssh or Windows. For more detailed instructions, go to <http://www.cs.washington.edu/education/courses/cse441/13sp/howtos.html>

As this is the first studio design assignment, we would like you to create a top level html page (index.html) in this directory that will refer to each of the studio assignments you are given in class. Please create this index.html and include a link to this week's studio design html document (described below).

2.

3. Your Design

Your design should be accessible online in your individual project directory as an html (or, if you prefer, pdf) page. This page should be linked off your index.html defined above. Please include screenshots of your sketches and a description of the interface and proposed interaction. Be sure to also bring a printout of this page to class.

4. Presentation

You must be prepared to present your work in class and lead a discussion about your design.

Grading (100 Points)

These studio assignments are meant to inspire your creativity, flex your design muscles, and give you a chance share your own designs in class. The assignments are meant to be rather lightweight and fun. It's also an opportunity to earn points—but only if you do the work.

For this assignment, you will be graded on:

- [10] Creating a top level index.html, which links to the first studio assignment
- [80] Your design in an html/pdf page
- [10] Your presentation