The Fractal Perspective Visualization Technique for Semantic Networks

Curran Kelleher, Georges Grinstein
University of Massachusetts Lowell
{ckellehe@cs.uml.edu, grinstein@cs.uml.edu}

Abstract

We introduce a novel interactive visualization technique for semantic networks supporting continuous semantic graph browsing. Our visual semantic graph representation is a nested object visualization representing a view into the graph from a node's viewpoint. In this view, panning and zooming interactions drive graph traversal. We call the visualization technique Fractal Perspective, because it has approximate fractal structure (self similarity at multiple scales) and represents a kind of "perspective projection" from graph space to display space in which distant objects (in graph space) appear smaller (in a dynamic display space). This is work in progress, and this paper should be considered primarily a concept paper with a brief description of a very early prototype harnessing multi-touch interaction.

1. Introduction

We address the problem of visually representing a semantic graph such that the graph is intuitively and efficiently browsable. One is tempted to imagine that the semantic graph data structure and related reasoning algorithms may partially approximate the way humans store memories and think. If this is even remotely true, an interactive visual tool representing semantic graphs offering intuitive interactions for data navigation (and ideally creation and manipulation as well) could serve as a revolutionary information recording, manipulation and communication technology.

Static visual representations of trees and graphs exist, as well as standards for representing semantic graphs, but these areas have not yet been combined to form a semantic graph browsing tool based on interactive data visualization concepts. There are some examples of dynamic representations in limited environments with an underlying semantic framework (see thebrain.com for example). However the semantic graph data structure, and thus the Semantic Web, still lacks a compelling and usable visual interface for data navigation, manipulation and creation. Were one created, it may serve as a catalyst for the adoption of the Semantic Web as a mainstream technology for personal and organizational information management, visualization and analysis.

We see the task of building visual interfaces for the Semantic Web as a significant step toward achieving several grand challenges, namely enabling memories for life, lifelong learning environments, and cognitive partners for humans. To this end, we propose a novel visualization technique for semantic graphs we call Fractal Perspective as a first stab at the problem.

2. Related Work

Our work combines elements of tree visualization and graph visualization. TreeMaps, introduced by Ben Shneiderman in [1], are visual representations of hierarchies whose nodes have associated numeric values. In a TreeMap, packed squares represent nodes, and color represents their associated numeric values. The Voronoi TreeMap, introduced by Balzer et. al. in [2], is a TreeMap variant where nodes are visually decomposed into Voronoi tessellations rather than packed squares. Circular TreeMaps have also been introduced by Wetzel [4], in which circles are the nesting unit.

Labeling of nodes in a TreeMap is only possible when overlaying the label to summarize a region (see Figure 1), or when enough padding is added such that labels can fit within the padding and be associated unambiguously with a particular node. Sometimes only the leaf nodes are visible in a TreeMap, but padding can be added to make all intermediate nodes visible. For our application, we prefer a nested object visualization technique which both allows unambiguous labeling and coloring of all intermediate nodes. As illustrated in Figure 2, the Circular TreeMap seems more amenable to unambiguous labeling and all-node coloring than the TreeMap or Voronoi TreeMap.

Researchers have produced many radial visual representations of a depth-limited or "node centric" tree view. In these representations, nodes are presented with varying levels of detail based on their distance from the root node, providing an effective focus plus context visualization. Interactive systems employing these visualizations have a browsing state which defines a particular node to be the root of the visualization at any given time. The mutability of this state via user interaction is the key element which enables interactive browsing of trees using these visualizations.

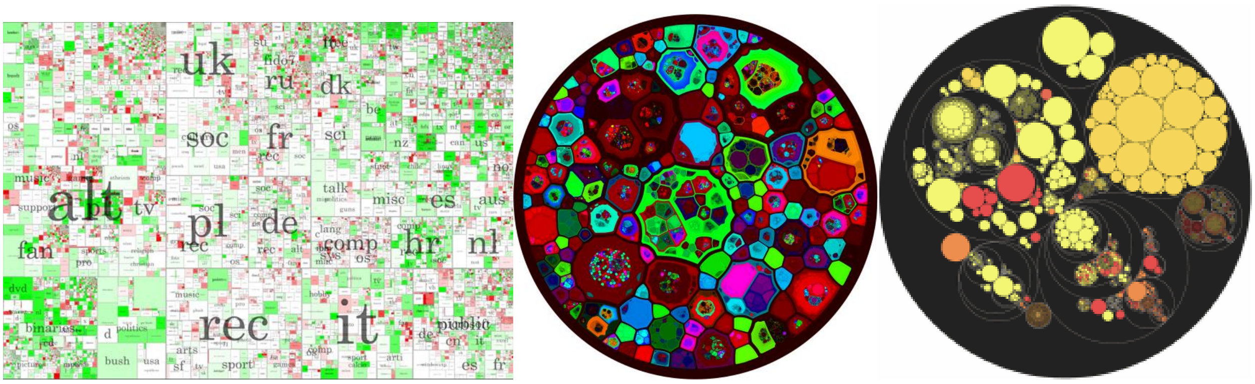

Figure 1. TreeMap, Voronoi TreeMap and Circular TreeMap. Images left to right from [3],[2] and [4].

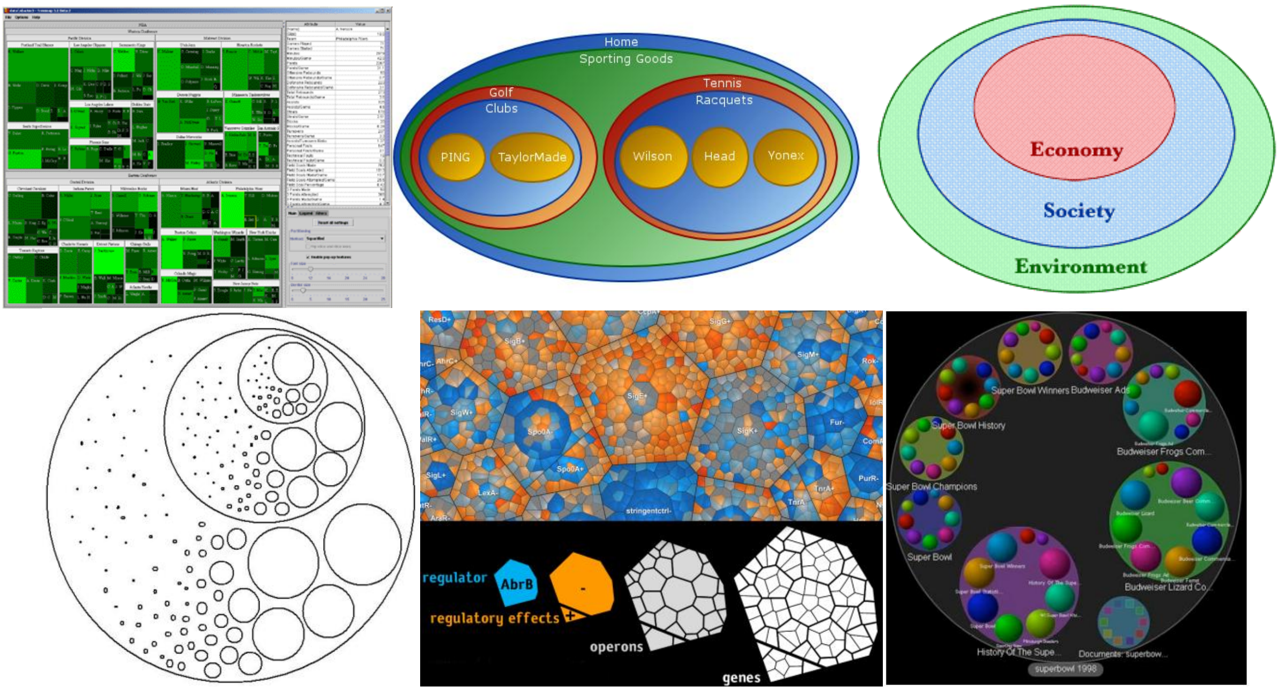

Figure 2. TreeMaps and variants used on semantic data spaces. Images left to right from (top) [1],[5],[6] (bottom) [9] – a Circular TreeMap visualization of Wikipedia structure,[8] – a Voronoi TreeMap gene ontology visualization, and [7] – a federated content search result viewer using nested circles.

Semantic graphs are directed graphs with labeled nodes and edges. Resource Description Framework (RDF) is a flexible data model for federated knowledge representation based on the semantic graph data structure. RDF is the foundation of the Semantic Web, Tim Berners Lee's vision for a global web interlinking data, computational agents and people, also called the "Network of Linked Data" or the "Giant Global Graph".

When confronted with the problem of visualizing semantic graphs, typically labeled node-link diagrams are used, with the distinction between Resources (object nodes) and Literals (data nodes) visually encoded (as in Figure 4). This approach does not scale, because as the size of the graph increases the size of the nodes and labels decreases, leading to unreadable visualizations.

We found only a few applications of visualization techniques to semantic data spaces in general. We were particularly impressed by the clear legend of Jörg Bernhardt [8](center of figure 2) depicting four hierarchical levels of a gene ontology including regulators, regulatory effects, operons and genes, with summary labels at the regulator level. Cartographic map navigation has many often noted advantages over other visualizations, the primary advantage being that it involves multi-scale generalization - including details appropriate to the scale at which the space is viewed. Panning and zooming over cartographic maps are the only two interactions necessary to explore the entire vast space represented by a multi-scale cartographic map. These interactions are made extremely intuitive by the (now conventional) multi-touch interactions of pinch zooming and drag panning. The addition of inertial processors to the multi-touch panning and zooming interaction make it even more instinctive (arguably because as humans we are used to manipulating masses with inertia).

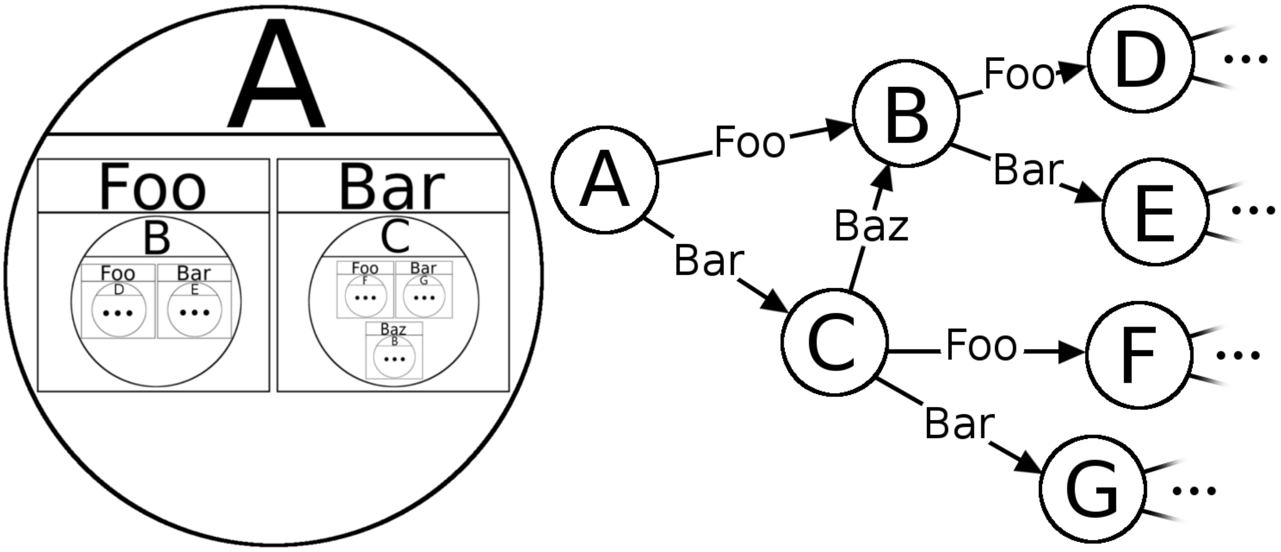

Figure 3. Fractal Perspective (left) and the corresponding node-link representation (right).

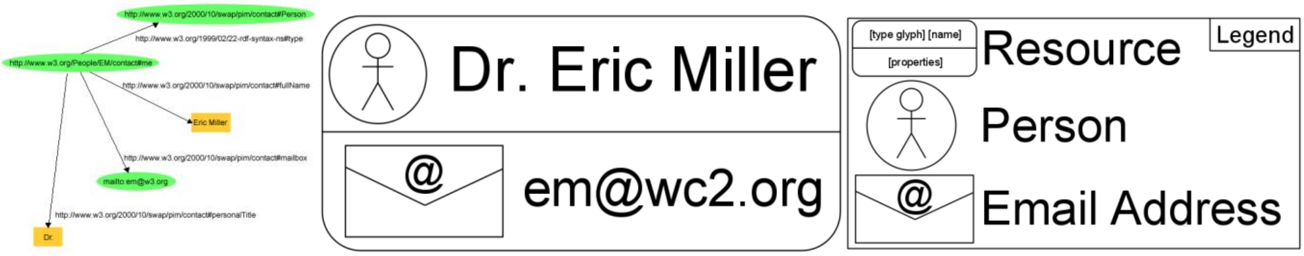
http://www.w3.org/2000/10/swap/pim/contact#Person
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/People/EM/contact#me
http://www.w3.org/2000/10/swap/pim/contact#fullName
Eric Miller
http://www.w3.org/2000/10/swap/pim/contact#mailbox
mailto:em@w3.org
http://www.w3.org/2000/10/swap/pim/contact#personalTitle
Dr.
Dr. Eric Miller
@
em@wc2.org
[type glyph] [name]
[properties]
Resource
Legend
Person
@
Email Address

Figure 4. An example ontology-specific visual language we'd like to enable users to define using the Fractal Perspective concept. The left image is a node link RDF visualization taken from [12]. On the right is a mock up visual language based on the Fractal Perspective concept which uses domain specific visual encodings to unambiguously and compactly visualize the same information represented in the node link diagram on the left.

2. Our Contribution

Our aim with this work is to apply the intuitive multi-scale navigation techniques used by interactive (particularly multi-touch) cartographic maps to the task of browsing a semantic graph. We also plan to apply this technique to browsing data cube structures. Though interactive navigation of hierarchical data cube dimensions was our original intent for this work, we address instead the problem of visually navigating semantic graphs because it has became clear in our research that the data cube structure itself can be represented within a semantic graph [10], and so solving the semantic graph navigation problem with sufficient flexibility implicitly solves the data cube navigation problem (or so we believe at this time).

We introduce a novel mapping between the data structure resulting from a rooted depth-limited semantic graph traversal and a nested object visualization. In our proposed interactive system, there is a mutable browsing state which consists of the root node for the visualization, a scaling factor (zoom), and a translation vector (pan). Multi-touch zooming and panning interactions affect the scaling and translation components of the browsing state while viewing children of a certain node. When the user zooms into a particular node so far that no part of the enclosing node is visible, the root of the browsing state changes to the newly focused node. At this point, a transformation of the scaling and translation components occurs such that the new browsing state results in the same visualization as the old one, just with a new root node (the scale and translation are projected from the space of the outer node to the space of the inner node). In this way, we establish an equivalence between browsing a multi-scale 2D space and browsing a semantic graph.

In Figure 6, circles represent windows through nodes, squares represent windows through edges, and ellipsis (...) represent nodes which have the potential for expansion. Nodes below a certain size threshold are left unevaluated, and are evaluated when the user zooms in on them (during retrieval, a progress animation is displayed in place of the ellipsis). In our proposed system, a force directed layout algorithm acts on the children of each node independently. Layout occurs on each set of child nodes independently, and the layouts of multiple child sets are combined during rendering using object containment.

2.2 Visual Languages for the Semantic Web

The basic specification of our Fractal Perspective visualization technique deliberately leaves out the details of how nodes and edges be represented visually. The only requirement is nesting. We view the visual language used in Figure 3 to be one of an infinite sea of potential visual languages for representing semantic graphs using the Fractal Perspective concept mapping connection to containment. Some of these may use explicit graphical objects for representing edges, some may express the edge label via some visual aspect of the node pointed to by that edge. We hope to enable the refinement of ontology-specific visual languages which have particular visual representations for particular ontology classes, properties and literals. For example, the fact that a Resource is of type Person could be represented by placing a stick figure next to the name of a node (as in Figure 4). As another example, an envelope glyph could be used to represent a person's email rather than a box labeled "Email". In Figure 4, the image on the left represents the traditional node-link semantic graph visualization technique used in [12]. The images to the right represent an ontology-specific visual language for describing people; a compact visual representation of the same semantic graph represented in the left side image.

In our proposed system, a visual language in which labeled circles represent nodes and labeled squares represent edges is taken to be the default. This is generic and can represent any semantic graph. We hope to enable users to progressively define ontology-specific visual languages by annotating each class and property with information specifying how to encode them visually. For a given class or property, the following can be specified: how is the label derived (e.g. from a class property, from a regex on the URI)? What shape should be used to represent a class or property instance? How should that shape be styled (color, texture)? What class properties drive which visual channels, if any?

We hope the rough conceptualization presented here can eventually lead to the formulation of a kind of "Grammar of Semantic Network Visualizations" built upon concepts from Wilkinson's Grammar of Graphics, Bertin's Semiology of Graphics, and the Semantic Web. A statement in such a grammar would be a formal version of the kind of legend shown in figure 4, describing exactly how elements in the semantic graph map to visual representations.

It is worth noting that our proposed visualization technique enables dynamic visualization of incremental (edge-by-edge) data requests. As more data is retrieved, the details of the visualization are filled in. This approach fits perfectly with the Linked Data paradigm [11] in which each node is identified by a URL, and outgoing edges for that node are retrieved in a group by accessing ("dereferencing") that URL. Our proposed system could enable navigation of Linked Data in a fluid manner in which requests can be literally seen coming in, causing incremental nested refinement of the visualization.

Our belief is that once ontology-specific visual languages are defined, Linked Data can be traversed and comprehended with ease by the masses using our technique. We plan to develop a prototype implementation for mobile multi-touch devices utilizing intuitive gestures for pan and zoom. We imagine that the more familiar one becomes with the various ontology-specific visual languages defined, the more quickly one is able to mentally register the semantic structure - the real meaning and implications - of the data encoded in the visualization. Our hope is that eventually, this technique will be used by many to communicate complex webs of meaning more effectively than existing techniques such as plain text, tree maps or node-link diagrams.

3. Future Directions

One disadvantage of our nested object approach is that nodes reachable by more than one path are represented more than once on the display. For example, node "B" is displayed twice in Figure 3: once within node "A" and once within node "C". In future work, we hope to address this issue by introducing visual elements which express equivalence between multiple representations of the same node by drawing a visual connecting element, and also perhaps to apply attractive forces which pull those nodes together to reduce clutter.

Imagine a physical setting in which several multi-touch tablets are arranged on tables surrounding a large high resolution display. On each tablet imagine a nested object view representing a window into a depth-limited graph neighborhood navigable via multi-touch panning and zooming. On the large display imagine a node-link diagram of the full graph, with a distortion bubble surrounding the locations of each nested object viewpoint. This would allow all nodes and edges seen by each viewer to also be seen by all in the room on the large display. At the same time, nodes outside the view of any user would be shrunk or represented by aggregate nodes on the large display. Each local nested object visualization is isomorphic to the contents of the distortion bubble surrounding its viewpoint represented in the node-link visualization. This is one potential direction we are considering, which would enable real time collaborative exploration of massive semantic graphs.

We would also like to explore how a system using this visualization technique can be built in which a small multi-touch device (such as an iPad) is used as a controller for a much larger display (such as a powerwall) showing the same view into the graph. In this scenario, the two devices must maintain a synchronized browsing state. The smaller device will only be able to display a few nesting levels, while the large display will look exactly the same as the small display, except that the smallest nodes in the small display could be filled in with several additional nested levels of detail.

4. Conclusion

We present the conceptual formulation of a novel visualization technique using nested graphical objects to facilitate visualization and navigation of semantic graphs. In the nested object view, some objects represent nodes and some represent edges, and each type of object is communicated using a clear visual encoding. We hope to evolve this work to enable user definable ontology-specific visual languages. Our approach has the disadvantage that nodes reachable via distinct paths are represented more than once in the display. We frame this work as a step toward building intuitive visual interfaces for the Semantic Web, which we see as ultimately a step toward achieving the grand challenges of enabling memories for life, lifelong learning environments, and cognitive partners for humans.

References

[1] Schneiderman et al. “Treemaps for space-constrained visualization of hierarchies”. url: www.cs.umd.edu/hcil/treemap-history/

[2] M. Balzer, O. Deussen. “Voronoi Treemaps”. IEEE Symposium on Information Visualization (InfoVis), 2005

[3] Treemap View of 2004 Usenet Returnees url: scimaps.org/maps/map/treemap_view_of_2004_57/

[4] K. Wetzel. “Pebbles - using Circular Treemaps to visualize disk usage”. url: http://lip.sourceforge.net/ctreemap.html

[5] Using the Nested Set Data Model for Breadcrumb Links. Jason Mauss. 2005. url: http://www.developer.com/db/article.php/3517366/Using-the-Nested-Set-Data-Model-for-Breadcrumb-Links.htm

[6] Wikipedia contributors. "Sustainability." Wikipedia, The Free Encyclopedia. 21 Mar. 2011.

[7] Rivadeneira, W., & Bederson, B. (2003). A study of search result clustering interfaces: Comparing textual and zoomable user interfaces. url: http://www.infostrategist.com/NavOpenSourceWeb/Oceania04_Images.html

[8] Jörg Bernhardt; Henry Mehlan; Julia Schüler; Michael Hecker. “Ontology Maps” (E.M.A.-University Greifswald, Institute for Microbiology; Jahnstrasse15;17487 Greifswald; Germany)

[9] Lizorkin, Medelyan, Grineva. “Analysis of Community Structure in Wikipedia” (Poster). 18th International World Wide Web Conference. 2009.

[10] Richard Cyganiak, Dave Reynolds, Jeni Tennison. "The RDF Data Cube vocabulary". url: publishing-statistical-data.googlecode.com/svn/trunk/specs/src/main/html/cube.html

[11] Tim Berners-Lee. "Linked Data". 2006-2009. url: www.w3.org/DesignIssues/LinkedData.html

[12] Frank Manola, Eric Miller, "RDF Primer". Web resource, url: www.w3.org/TR/rdf-syntax/