Journal of Universal Computer Science, vol. 16, no. 7 (2010), 1036-1054
submitted: 19/1/10, accepted: 29/3/10, appeared: 1/4/10

Ontology Visualization: Tools and Techniques for Visual Representation of Semi-Structured Meta-Data

Monika Lanzenberger[1]
(Vienna University of Technology
Favoritenstr. 9-11, 1040 Vienna, Austria
lanzenberger@ifs.tuwien.ac.at)

Jennifer Sampson
(Statoil, 5020 Bergen, Norway
jensam@statoil.com)

Markus Rester
(Vienna University of Technology
Favoritenstr. 9-11, 1040 Vienna, Austria
markus@igw.tuwien.ac.at)

Abstract: Ontologies are used to represent a variety of domain knowledge and data collections, scopes, viewpoints and linked heterogeneous information sources. They range from simple topologies to highly structured knowledge bases with complex relations. When mapping or aligning two or more ontologies an efficient user support is needed so that the users can understand the prerequisites and the consequences of the alignments. Information Visualization techniques can help to facilitate user understanding of the ontology alignment results. In general, a lot of work in visualization of ontologies exist. We found an enormous number of ontology visualization tools by a literature study. Many of them apply graph visualization but there are other approaches as well. We have identified interesting solutions for dealing with the complexity of large ontologies. Ontology engineering, ontology mapping and alignment can benefit from Information Visualization. Our collection is a starting point to demonstrate the usefulness of Information Visualization techniques, however, a detailed evaluation would be the next step to consolidate this research area and help to boost the adoption of ontologies in common Web applications.
Key Words: Semantic Web, Ontology, Information Visualization Techniques
Category: H.1.2, I.2.m, I.6.3

1 Introduction

Visualization has an appealing potential when it comes to creating, exploring or verifying complex and large collections of data such as ontologies. In particular, Information Visualization (InfoVis), which deals with abstract and non-spatial data, offers a bundle of techniques to represent hierarchical or semi-structured data. So, it is no surprise that many ontology tools integrated visualization in

[1] Current Address: European Research Council, ERCEA, Covent Garden 21/23, Place Rogier, B-1049 Brussels

one way or the other during the last decade. Many tools rely on simple types of visualizations like two-dimensional trees or graphs. Usually the nodes stand for concepts and the edges represent relationships between concepts. However, some other approaches do exist as well. A literature study indicated a broad interpretation of ontology visualization differing among the various tools. InfoVis uses visual metaphors to ease the interpretation and understanding of multi-dimensional data in order to provide the user with relevant information. Graphical primitives such as point, line, area or volume are utilized to encode information by position in space, size, shape, orientation, color, texture, and other visual cues, connections & enclosures, temporal changes, and viewpoint transformations [Card et al., 1999]. The goal of InfoVis is to promote a more intuitive and deeper level of understanding of the investigational data and to foster new insights into the underlying processes [Tufte, 2001]. An enormous amount of work has been done in the field of InfoVis during the last years. The methods range from geometric techniques (e.g., scatterplots and parallel coordinates [Inselberg and Dimsdale, 1990]), Glyphs (e.g, InfoBug [Chuah and Eick, 1997]), icon-based techniques (e.g., Chernoff faces [Chernoff, 1973], stick figures [Pickett and Grinstein, 1988]), pixel-oriented techniques (e.g., recursive patterns [Keim et al., 1995], spiral- and axes techniques [Keim, 1996]), to interactive visualizations for hierarchical information (e.g., cone or cam tree [Robertson et al., 1991], hyperbolic tree [Lamping et al., 1995]), graph-based techniques (e.g., small world graphs [van Ham and van Wijk, 2004]), maps (e.g., themescape [Wise et al., 1995]), distortion-oriented (e.g., fisheye lens [Furnas, 1986]) and other focus+context techniques [Pirolli et al., 2001] or hybrid techniques (e.g., Stardinates [Lanzenberger et al., 2003]). Combining several views is well-known as multiple view visualization which offers a lot of advantages, such as improved user performance, discovery of unforeseen relationships, and unification of the desktop [North and Shneiderman, 1997]. Generally, in InfoVis the exploration process is characterized by cognitive abstraction. In addition, visualization itself often reduces information or emphasizes certain aspects of the data in order to ease goal-oriented interpretation. Combining distinct visualizations yields different kinds of abstractions from the data, which allow for diverse approaches of exploration. An important challenge of multiple view visualization is its complexity for the users. They need to switch between different views and contexts.

In the following we shortly describe the relevant ontology tools, which exploit InfoVis techniques. However, more tools not listed here may exist as well.

2 Graph-based visualization tools

Ontorama [Eklund et al., 2002] is a hyperbolic-style browser designed to render RDF files derived from a web-accessible ontology server called WEBKB-2 [Martin and Eklund, 1999], which contains descriptions of over 74,500 object types from WORDNET [Miller et al., 1990]. Ontobroker [Decker et al., 1999] utilizes a hyperbolic tree view as well. Ontobroker is an ontology-based semantic indexing and instance querying technology for the WWW. Detail and overview is implemented by the ISWIVE tool [Chen et al., 2005], which aims at integrating RDF and Topic Maps. The user can decide whether to view the RDF or the Topic Map graph in the so-called dual viewer providing the overview. The local viewer visualizes selected nodes in more detail, e.g. the labels of objects and attributes. Both views are linked together, forming a multiple view tool. A multi-scale force-directed algorithm is used to arrange the RDF graph; the Topic Map view exploits another standard algorithm to avoid crossings. The tool GViz [Telea et al., 2003] is suitable for larger RDF graphs. A nested layout, a 3D stacked layout, and a spring embedder-based tree are used to visualize the graphs. Moreover, GViz allows for customizing the visualization's appearance and supports the exploration process by flexible scenario-specific queries. FOAFnaut [FOAFnaut] employs combined 3D graphs to visualize the FOAF (friend of a friend) [FOAF] data. Each person of this network is represented by a node. Friends are connected by edges showing the social structure in this community. Several different types of 2D graphs, e.g., drawn by RDFAuthor [Steer], are used to visualize the relations of companies with the FOAFcorp [foafCORP] similar to the friendships of people. Similarly the Flink system [Mika, 2005] uses a graph to visualize social networks derived from a number of electronic information sources including web pages, emails, publication archives and FOAF profiles using RDF data. Flink visualizes the Semantic Web research community showing main topics of the field and the actors.

The Co4 system [Euzenat, 1996b] aims at the representation and integration of corporate memory, which can be seen as formal and informal knowledge. Visualization of Co4 system is done by HyTropes [Euzenat, 1996a, Jung, 2008b], which creates simple graphs to display the concepts and their relations. WebODE [Arpírez et al., 2001] uses the tool called OntoDesigner to graphically edit ontologies using common node/edge to represent the concepts and the relations in a tree. Tadzebao [Domingue, 1998], which is a tool for collaborative development of ontologies, includes the tree-tool WebOnto for the same purpose. Also FCA [Stumme and Maedche, 2001] uses simple node-link visualizations of the inherent structure. Conzilla [Naeve, 2001] and VizCo [Fuchs et al., 2003] apply RDF-graphs to create and manipulate ontologies. Vizigator [Gennusa, 2004] represents topic maps using the Touchgraph technology [Touchgraph]. ViSWeb [Dori, 2004] is an OPM-based (Object-Process Methodology) layer on top of XML/RDF/OWL to express knowledge visually and in natural language. Both, the visual and the textual representation are strictly equivalent. It employs a graph layout claiming to be both, intuitively understandable by humans and processable by computers. ORIENT (Ontology engineering Environment) [Apex Lab] is an Eclipse-based system using RDF-graphs and includes ontology building, mapping, evolution, evaluation and visualization. RDFAuthor [Steer] supports the creation of RDF instance data by dragging the data into a graph and binding it together using a graphical and quite simple interface. IsaViz [Pietriga] relies on GraphViz [Gansner and North, 2000] to browse and author RDF models presented as graphs. Similar is the FRODO RDFSViz tool [Sintek], which provides class models of ontologies represented in RDF Schema using GraphViz. Another 2D graph visualization is StarLion [Tzitzikas et al., 2007] which is part of the RDFSuite. It is a graphical editor for visualizing RDF schemas. It is also based on a force-directed placement algorithm. In order to avoid an overwhelming number of edge crossings, which is a typical problem with such graphs, this approach uses a ranking method. Thus the graph shows only a part of the RDF Schema and allows for exploring it gradually. This smaller (top-k) diagram shows only k elements from a ranked list considered as more important than the rest. For social networks based on RDF Data, such as FOAF, [Jung, 2005, Jung, 2008a] has develped a graph visualization approach for recommender systems. This technique offers visual explanation of the predicted recommendation on a social network. With this visualization the user is able to see and understand specific relations among social actors quickly.

Building ontology-based queries with different levels of guidance is the aim of GODE (Graphical Ontology Design Environment) [Wienhofen, 2004]. This tool lets the user organize her or his concepts in three different areas: a main concept area, which contains the core search concepts; an area for background knowledge concepts; and an area for temporary storage of concepts. The user can also do textual (guided) search: based on the query text a ”query ontology” is built by OntoExtract [Engels and Bremdal, 2001] which is transformed by GODE to a graphical output using the spring embedder algorithm. Microsoft's Visio builds the environment for the graphical representations of the tool VisioOWL [Flynn]. For Description Logics we mention InfoLens [Ng, 2000]. It is a visualization environment for GRAIL (GALEN representation and integration language) modeling [Rector et al., 1997] which adopts the paradigm of magic lenses as a hybrid of the data-flow network and spreadsheet paradigms. Lenses are used to formulate and manipulate object collections, and to present comparison matrices. A number of visual interfaces exist for the knowledge base Cyc [Lenat and Guha, 1989] Two early tools are MUE (Museum Unit Editor) [Travers, 1989] which is a visual browser based on nested boxes and color encoding using a room metaphor and VISAR [Clitherow et al., 1989] using a graph metaphor. The more recently developed Webstructor [Kolonin] can be used as visual graph-based front-end for Cyc. The Smalltalk tool CODE4 (Conceptually Oriented Description Environment) [Skuce and Lethbridge, 1995] provides node-link diagrams and table-based views: the so-called property matrix enables the user to compare the properties of two or more concepts. This approach influenced the GKB-Editor [SRI International] another authoring and browsing tool for knowledge bases. It provides a tree view where the frames and concepts are nodes and the slots and values are edges. The main views are the class hierarchy, the frame relationships viewer, and the frame editor. If a frame has multiple super-classes, an edge is drawn from each super-class.

3 Protégé plug-ins

A number of graph-based visualization plug-ins exist for Protégé [Grosso et al., 1999]. The OntoViz tab [OntoViz] visualizes the Protégé ontologies with the graph drawing software GraphViz and looks similar to IsaViz. OntoViz is useful for graphically representing small ontologies or ontology fragments. Another plug-in called Jambalaya [Storey et al., 2001] uses SHrIMP [Storey and Muller, 1995] to visualize ontologies and adds in a zooming feature. It uses two different visualization metaphors: one the one hand, a classical tree layout, on the other hand, nested view visualization. Compared to OntoViz it is able to deal with more complex ontologies. Using TouchGraph [Touchgraph], the TGViz [Alani, 2003] plug-in provides the user with a vivid impression of the taxonomic structure by making use of an animated spring-embedder algorithm. The user can expand nodes easily in order to view more details of the graph. OWLViz [Horridge] is a graph-based tool as well, which enables the user to compare the asserted and the inferred class hierarchy. Colors indicate differences in both class hierarchies and inconsistent concepts are highlighted. The tool allows for saving the graphs in different file formats. ezOWL [Electronics and Telecommunications Research Institute] supports modeling visually by manipulating a graph representation of the ontology. The OntoSphere 3D tool [Bosca and Bonino, 2005] uses several visual cues besides the 3 spatial dimensions: color, shape, and transparency encode concepts' features such as taxonomic relations, concepts versus instances, and inherited / inferred versus direct relations. For dealing with large ontologies this tool includes hyperbolic distortion techniques which enables the user to view parts of the ontology in detail while others are zoomed out. It applies scene managers, e.g., the so-called RootFocus or the TreeFocus to ease the exploration of ontologies.

4 Visualization for mapping and alignment

A few visualization tools support the user with ontology mapping and alignment. We have identified four such tools, some of them are Protégé plug-ins as well. OLA (OWL Lite Alignment) [Euzénat et al., 2004], a stand-alone program, uses

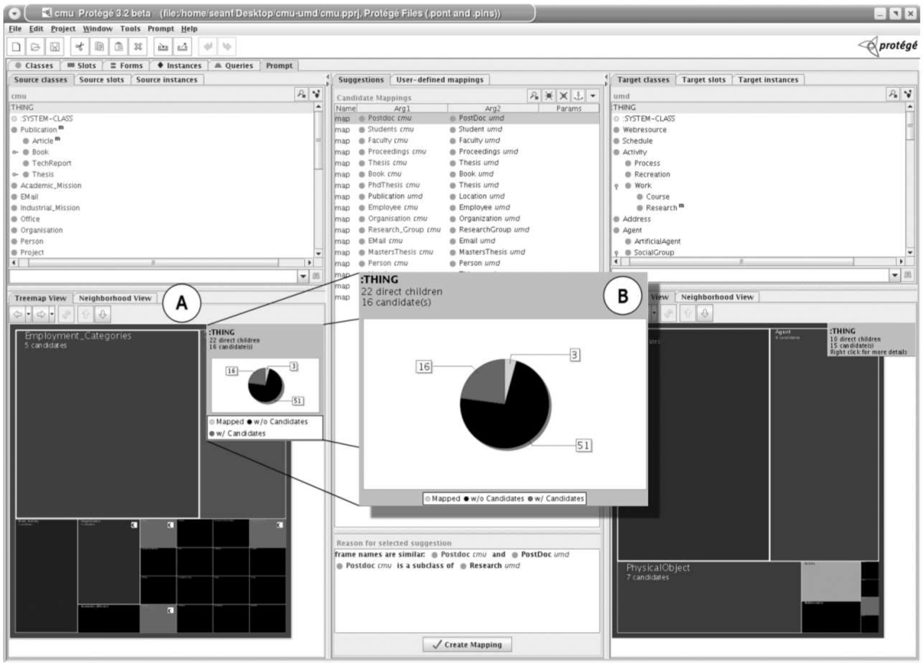

Figure 1: The CogZ Tool showing the candidate mappings and the actually mapped subclasses of owl:Thing.

graph-based visualizations to represent ontologies. In particular, an extended JGraph API is applied. The graph structure of OLA makes relationships between language elements explicit, e.g., if a class `c` refers to another class `c'` via a `owl:allValuesFrom` restriction, a labeled path between the corresponding nodes in the OL-graph is shown such that the connection between both classes is perceived intuitively. Besides common sub-class relationship the user can activate the display of edges between objects that are reverse, symmetric or transitive. Coma++ [Aumueller et al., 2005], a stand-alone tool for schema and ontology matching, uses simple lines to connect the mapping pairs in list views. However, its main focus is the mapping algorithm, not the visualization of the mapping results.

PromptViz [PromptViz] is a visualization tool for Protégé's Prompt tool [Noy, 2004]. It provides visual representations of the differences between two versions of an ontology using histograms within a treemap. The bars in the histograms represent the percentage of descendents classified as unchanged, added, deleted, moved-from, moved-to and directly changed respectively. It is divided into four linked frames: (1) an expandable horizontal tree layout of the ontology showing the differences; (2) a treemap layout of the ontology embedded in a zoomable user interface; (3) a path window that shows the location of the currently selected concepts within the is-a hierarchy of the ontology; (4) a detailed list of the changes (if any) that have occurred to the currently selected concept. Implemented as a user interface plugin extension to Prompt the tool CogZ [Falconer and Storey, 2008] offers visual mapping functionality. It enables the user to examine, add or remove temporary or permanent mappings. A bundle of filtering options help to handle the complexity of ontology mappings. Moreover, neighborhood graphs and fish-eye lenses and other treemap or pie chart views offer efficient means for exploration of the mappings. Figure 1 shows the CogZ tool with the mapping results derived from Prompt.

AlViz [Lanzenberger and Sampson, 2006] is a research prototype for visual ontology alignment implemented as multiple-view plug-in for Protégé (see Figures 2 and 3). Based on similarity measures of an ontology matching algorithm AlViz helps to assess and optimize the alignment results on different levels of detail. Clustered graphs enable the user to examine and manipulate the mappings of large ontologies. More details about AlViz are described in the next subsection.

4.1 AlViz - Multiple-View Visualization for Semi-Automatic Alignment of Ontologies

AlViz supports the visual alignment of two ontologies by making the type of similarity between entities explicit. The tool consists of two types of views coupled by the linking and brushing technique. AlViz applies J-Trees as one out of two types of views. Such trees consist of a root node, expanded or collapsed branch nodes and leaf nodes displaying the hierarchical structure by indentation. They support the access and manipulation of instances and single items within classes quite effectively and are well established within the Protégé community. But such J-Trees bear shortcomings regarding the representation of large or complex ontologies because they become cluttered and do not provide adequate overview functionality. To overcome this problem another visualization type had been integrated: small world graphs [van Ham and van Wijk, 2004]. Therefore, as a second view, such graphs help the user to examine the structure of the ontology intuitively. This method uses clusters to group the nodes of a graph according to the selected level of detail. The nodes represent the entities (concepts or instances) connected to each other according to the selected relations, also called mutual properties, such as `IsA`, `IsPart`, `IsMember`, `locatedIn`, `IsDistinct`. So, each source ontology is visualized as a clustered graph where the edges represent the selected mutual property (or a cumulation of properties is possible as well).

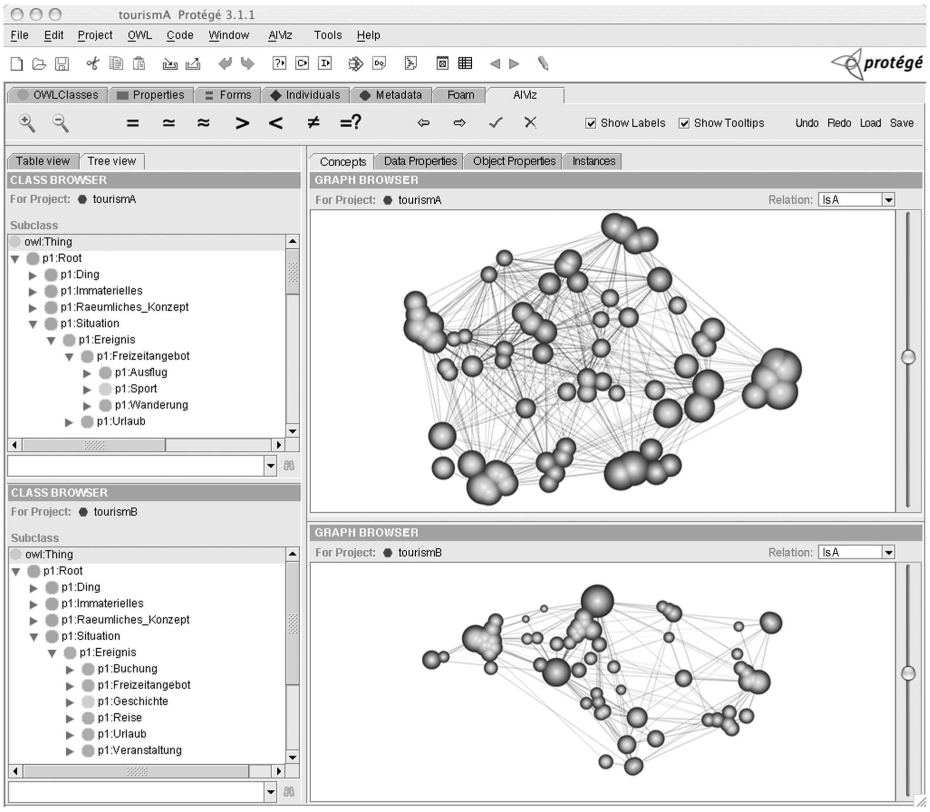

Figure 2: AlViz: the four views of the tool visualize two ontologies named tourismA and tourismB. The nodes of the graphs and dots next to the list entries represent the similarity of the ontologies by color. The size of the nodes results from the number of clustered concepts. The graphs show the `IsA` relationship among the concepts. Green indicates similar concepts available in both ontologies, whereas red nodes represent equal concepts. The sliders to the right adjust the level of clustering.

When aligning ontologies visually the users are interested in the global properties of the data such as: Are there any distinct groups of items that are strongly interconnected (i.e., graph clusters)? How do these split into separate clusters and how do these clusters relate? Small world graphs originating from [Milgram, 1967] in social networks are graphs, which have a small average path length (average shortest path between nodes) compared to their number of nodes, but have a high degree of clustering compared to a random graph of the same size.

Informal evaluation of structures of different ontologies showed the same property, which makes clustering a suitable approach. An adequate visualization for clusters should communicate the underlying structure efficiently. The number of visible elements in the graph needs to be reduced, but at the same time we shall maintain the global structure of the graph. That means to find a representative visual abstraction of the original graph. Although the small world graphs like all spring-embedded algorithms bear the problem of high computational complexity - usually $O(N^3)$, clustering the graph improves program's interactivity. The tool is fast enough to perform at interactive speeds because on average there are only $O(Log(N))$ clusters visible. The current implementation manages up to about 1000 entities per ontology. There is ongoing research [Newman, 2004, Jung, 2009b] to reduce complexity of such graphs, which looks quite promising for visualizing graphs with some ten thousands of nodes.

Each ontology is visualized by both views, the J-Tree and the small world graph, resulting in four linked views making the ontologies available to the user on different levels of detail-and-overview. Figure 2 shows AlViz visualizing the alignment of two ontologies about tourism with a high degree of details. Clustering the nodes like depicted in figure 3 the visualization emphasizes the hierarchical structure in the graph. Both figures visualize the same source ontologies on different levels of detail. The colors of the nodes and the dots next to the entities' names represent the degree of similarity or type of association respectively. Six categories of association between entities are distinguished: equal, syntactically equal, broader-than, narrower-than, similar and different. The input file representing these associations is discussed in more detail later on in this subsection. By default an entity of one ontology equal to an entity in the other ontology is colored red; a syntactically equal entity is colored orange; an entity broader than an entity in the other ontology appears blue; a narrower entity is colored violet, a similar entity is colored green; and finally, an entity different from all entities in the other ontology is colored yellow. In the graphs the clusters of nodes inherit the colors of the underlying nodes in accordance to the selected comparison strategy. Three different comparison strategies are available: the first one emphasizes similarity of entities, the second highlights differences among entities, and the third represents the entities according to the dominant type of associations. If the user focuses on similarity, the color of nodes indicating more similarity has priority against the color of nodes with less similarity. Particularly, clustering an equal and a syntactically equal node into one single node results in a red colored cluster. Of course, there are also higher levels of detail possible, maintaining two differently colored nodes as individuals but combined within one cluster. However, this case is simple. But if the cluster is represented by just one node, a priority approach is necessary.

Interacting with the graph involves a number of possible tasks. Beside the zooming functionality, a selection/highlighting function, several alignment functions, and tracking buttons for questionable associations are implemented. Selecting a node with a mouse click results in highlighting three other items. In particular, the entity (or group of entities) of the same ontology is highlighted in the tree view. While navigating the nodes (entities) in ontology O_1, the associated nodes in O_2 are highlighted. This interaction technique is known as linking and brushing. If the association is equality then the equal entity (or group of entities) in the other ontology is highlighted as well. This results in linking together entities between both ontologies, O_1 and O_2, in both views the graphical (over)view and text-based (detailed) view. The same holds for the other types of association, i.e., syntactical equality, broader-than and narrower-than relations, and similarity. The alignment functions allow for adapting the automatically derived associations by assigning the type of association manually and thus approving or rejecting the alignment result. The user can select an entity in the graph or in the text list. Then she or he activates the required type of association, e.g., equality, followed by a selection of the associated entity (or group of entities) within the other ontology. Such changes of the alignment are rather complex. However, the interactive manner of the graph visualization makes this task easier and more manageable to the user. Undo and redo functions combined with a history of applied interactions, labeling and tooltips are included in the tool. AlViz supports the user in understanding the alignment process and manipulating its results in order to improve and maintain its quality. In terms of perception such small world graphs utilize features of the Gestalt tradition. In particular, the Gestalt Law of Organization: 'When we are presented with a number of stimuli, we do not as a rule experience a number of individual things ... Instead larger wholes ... are given in experience, their arrangement and division are concrete and definite' [Wertheimer, 1967]. The Gestalt psychologists thought that the way in which parts of a figure are influenced by such factors as proximity, good continuation, and closure, reflects a natural tendency toward good forms in our experience [Coren and Girgus, 1980, Jung, 2009a]. Beside features as the lie position, size, and shape the color is an outstanding property, which supports the perception of items belonging together. Another aspect is explained by Asch's Concept of Unit Formation: It is easier to remember two or more objects, if you see them as one unitary pattern. Gestalt psychologists believed the reason is that unitary patterns are much more coherent perceptually, and can be remembered as units. When switching among different levels of clustering the user perceives the various groups of nodes as units, which helps to remember the position of the individual entity. However, a detailed discussion of the perceptional features of AlViz exceeds the space of this paper.

For the calculation of the alignment values an extended version of the general alignment process of [Ehrig and Sure, 2004] had been used. This process includes

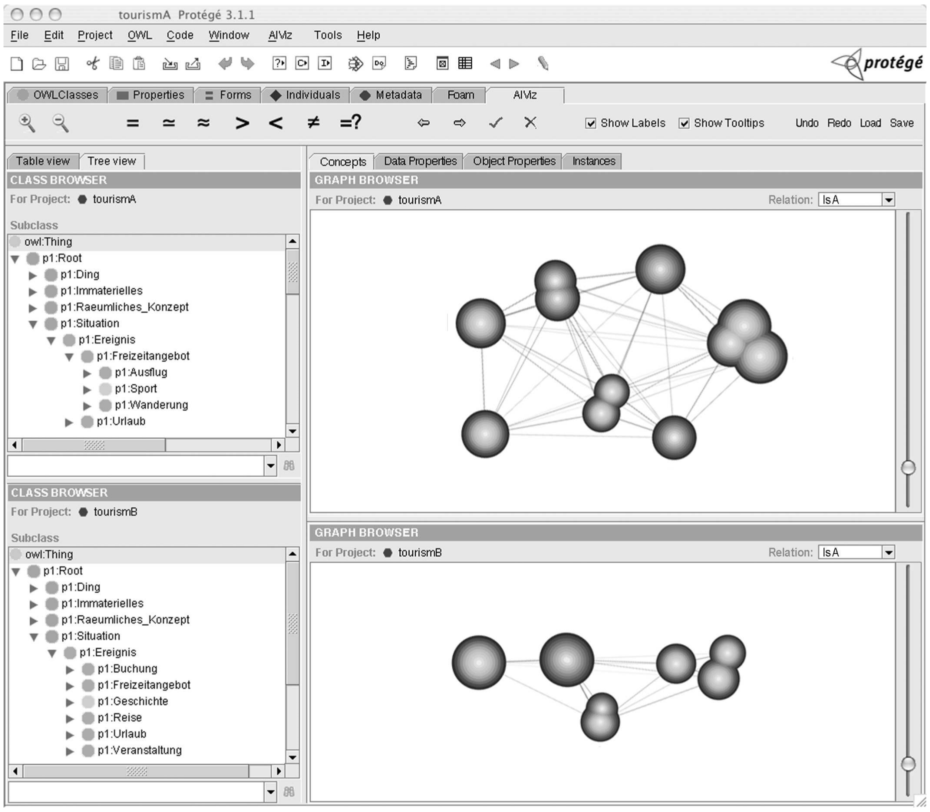

Figure 3: AlViz showing a highly clustered view of the alignment results for ontologies tourismA and tourismB. Different concepts are colored yellow representing entities which occur in only one of the two source ontologies. Orange represents syntactically equal concepts.

transformation, evaluation and visualization of the output results. Foam [Ehrig, 2005] an ontology alignment algorithm for generating candidate alignment relations between entities in two ontologies had been the basis in order to generate the input files using XML Schema to store the alignment data. The Foam algorithm uses a large number of rules for calculating similarity between entities. The reader may refer to [Ehrig et al., 2005] for similarity rules applied in Foam. The resulting similarities or associations between entities provide evidence that two entities are the same (or similar) and can potentially be aligned. The similarity function combines different features of the ontologies with appropriate similarity measures. As well as testing for equality, different similarity measures are used to calculate how similar entities in two ontologies are. The output from the Foam algorithm in a text file describes: `entity 1`, `label 1`, `entity 2`, `label 2`, overall confidence, individual rule and each correct value. The overall confidence represents the aggregated individual similarity values, the individual similarities correspond to the different ontology features which are further categorized in the six groups listed above (equal, syntactically equal, similar, etc.). The correct value corresponds to whether the alignment is correct or not with respect to identity. After generating the input file the user can select a property using a drop-down list, which shows all available relations (i.e., mutual properties) of the ontologies involved. Based on the selected property, the OWL source files of both ontologies, and the input file holding the categories of associations AlViz generates the visualization file. Finally, this file is used as input to draw the nodes and edges of the graphs and the entries of the J-Trees.

5 Ontology tools applying unconventional visualization techniques

In this section we list various ontology tools, which apply non-graph-based Info-Vis techniques, and therefore, are rather unconventional in the context of ontology visualization. The cluster map [Fluit et al., 2004] is a visualization technique suitable for light weight ontologies using RDF. It visualizes simple schemas with instances (shown as spheres) and scales to a large number of instances as well. Instances with the same class membership are grouped in clusters. When subclass relations hold between two classes, the clusters are connected by a directed edge. Cluster maps contain mainly information about the instantiation of the classes, specifically exploiting the overlaps between them. This technique is applied in a number of tools i.e. Autofocus [Fluit et al., 2005], Spectacle [Fluit et al., 2004], the DOPE project [Stuckenschmidt et al., 2004], and SWAP [Broekstra et al., 2003]. The visualization of the instances and their relationships is powerful for flat ontologies with a low number of overlaps among classes because too many crossing edges lead to a cluttered view. KNAVE (Knowledge-based Navigation of Abstractions for Visualization and Explanation) [Shahar and Cheng, 2000] is a domain-independent framework, which enables the user to interactively explore and interpret time-oriented data organized in an ontological structure. The main application field of KNAVE is to support clinicians to explore protocol-based recommendations. KNAVE applies simple timeline visualizations.

So far WebTheme [Whiting and Cramer, 2002] visualizes HTML-Websites only, although future developments may also work with OWL. We include this tool because it represents a rare visualization approach. WebTheme provides two types of images: the ThemeView visualization shows themes of the document space as a relief map of natural terrain, where taller peaks indicate important themes, on the one hand. The Galaxy visualization represents the document space with the individual web pages as stars, on the other hand. The distances among the stars indicate the degree of similarity. This is an interesting approach for visualizing overviews.

The SEWASIE project [Becks, 2003] applies a Kohonen-based clustering map combined with textual views to visualize documents (main focus is RDF and XML data). LexoVis [Pretorius, 2004], a lexon visualization tool provides overview and detail by employing the fisheye view resulting in different levels of detail. It assumes ontologies as sets of binary fact types - the lexons. This assumption limits its applicability for ontology alignment. Three clustering heuristics are implemented to organize the data in space: clustering by context, by terms, and by roles. Currently, the tool can represent 500 to 1000 lexons.

The idea of the Matrix Browser [Ziegler et al., 2002] is to map the underlying graph structure to a highly interactive adjacency matrix, where the nodes are shown on the horizontal and vertical axes. The cells of the matrix are marked if the two correlative nodes are connected by an edge.

6 Conclusions

We summarize the results of our literature study: A huge amount of tools exist for visualizing ontologies, however, there are just a few for assisting with viewing multiple ontologies as needed for ontology alignment. Although some tools emphasize intuitive user interfaces most tools require ontology expert knowledge. The time complexity of the layout algorithm needs to be low to maintain good interaction for the users and the layout needs to make efficient use of the screen 'real estate' so large numbers of concepts can be displayed simultaneously. Finally, in order to support an overview and detail approach appropriately, multiple views or distortion techniques are needed. Some tools show interesting and appropriate approaches to solve this problems. However, the killer application in ontology visualization is still missing. [Blundell and Pettifer, 2004] identify the following requirements for tools applicable to visualizing ontology changes and maintenance:

- Representation of the data transformed between two ontology versions.
- Informing the user of axiom validity across changes.
- Distinguish semantic and syntactical changes.
- Highlight potential specification violations resulting from semantic changes.
- Support basic and complex change distinctions / abstractions.
- Ability to vary the level of granularity of changes / group changes.

- Present data in an orderly fashion, reducing unnecessary detail, particularly in large ontologies.
- Related scope and abstraction.
- Ability to reverse changes and review previous ontology versions.
- Identify terms, which may become redundant due to changes.
- Differentiate between inferred and actual links.
- View the ontology with respect to different link types.
- Identify the term migrations performed by the reasoner.
- Explore the view interactively in order to discover further ramifications.
- Differentiate between original and imported classes / ontologies.

Based on this requirements a detailed evaluation of this enormous amount of ontology visualization tools and approaches would help to consolidate this research area and boost the adoption of ontologies in common Web applications. Moreover, visualization in ontology alignment allowing for intuitive user interfaces would foster integrating heterogeneous information sources.

References

[Alani, 2003] Alani, H. (2003). TGVizTab: An ontology visualization extension for protégé. In *Proc. of Knowledge Capture (K-Cap'03) – Workshop on Visualization Information in Knowledge Engineering*, Sanibel Island Florida.

[Apex Lab] Apex Lab. Orient. `http://apex.sjtu.edu.cn/projects/orient/index.htm`.

[Arpírez et al., 2001] Arpírez, J. C., Corcho, O., Fernández-López, M., and Gómez-Pérez, A. (2001). Webode: a scalable workbench for ontological engineering. In *Proc. of the 1st Intl. Conf. on Knowledge Capture (K-CAP '01)*, pages 6–13, New York, NY, USA. ACM.

[Aumueller et al., 2005] Aumueller, D., Do, H.-H., Massmann, S., and Rahm, E. (2005). Schema and ontology matching with coma++. *SIGMOD 2005*.

[Becks, 2003] Becks, A. (2003). Sewasie project: Specification of the interface design (ist-2001-34825 d4.4). Technical report, IST-2001-34825 D4.4.

[Blundell and Pettifer, 2004] Blundell, B. and Pettifer, S. (2004). Requirements for the visualisation of ontological evolution. In *Proceeding of Theory and Practice of Computer Graphics (TPCG04)*, pages 18–23.

[Bosca and Bonino, 2005] Bosca, A. and Bonino, D. (2005). Ontosphere: more than a 3d ontology visualization tool. In *SWAP 2005, the 2nd Italian Semantic Web Workshop*. `http://ceur-ws.org/Vol-166/70.pdf`.

[Broekstra et al., 2003] Broekstra, J., Ehrig, M., Haase, P., Harmelen, F. V., Sabou, M., Siebes, R., Staab, S., and Tempich, S. C. (2003). A metadata model for semantics-based peer-to-peer systems. In *Proc. of the WWW'03 Workshop on Semantics in Peer-to-Peer and Grid Computing*.

[Card et al., 1999] Card, S., Mackinlay, J., and Shneiderman, B., editors (1999). *Readings in Information Visualization.* Morgan Kaufman.

[Chen et al., 2005] Chen, I.-X., Fan, C.-L., Lo, P.-H., Kuo, L.-C., and Yang, C.-Z. (2005). ISWIVE: an integrated semantic web interactive visualization environment. In *Proc. of 19th Intl. Conf. on Advanced Information Networking and Applications, 2005 (AINA 2005)*, volume 2, pages 701–706.

[Chernoff, 1973] Chernoff, H. (1973). The Use of Faces to Represent Points in k-dimensional Space Graphically. *Journal of the American Statistical Association*, 68:361–368.

[Chuah and Eick, 1997] Chuah, M. and Eick, S. (1997). Glyphs for software visualization. In *Proc. of the Fifth Intl. Workshop on Program Comprehension (IWPC'97)*, pages 183–191.

[Clitherow et al., 1989] Clitherow, P., Riecken, D., and Muller, M. (1989). Visar: a system for inference and navigation of hypertext. In *Proc. of the 2nd ACM Conf. on Hypertext*, pages 293–304, New York, NY, USA. ACM.

[Coren and Girgus, 1980] Coren, S. and Girgus, J. (1980). Principles of perceptual organization and spatial distortion: The gestalt illusion. *Journal of Experimental Psychology*, 6.

[Decker et al., 1999] Decker, S., Erdmann, M., Fensel, D., and Studer, R. (1999). Ontobroker: Ontology based access to distributed and semi-structured information. In *Database Semantics: Semantic Issues in Multimedia Systems*, pages 351–369. Kluwer Academic Publisher.

[Domingue, 1998] Domingue, J. (1998). Tadzebao and webonto: Discussing, browsing, and editing ontologies on the web. In *Proc. of the 11th Knowledge Acquisition for Knowledge-Based Systems Workshop.*

[Dori, 2004] Dori, D. (2004). Visweb - the visual semantic web: unifying human and machine knowledge representations with object-process methodology. *The VLDB Journal*, 13(2):120–147.

[Ehrig, 2005] Ehrig, M. (2005). Foam - framework for ontology alignment and mapping. http://www.aifb.uni-karlsruhe.de/WBS/meh/foam/.

[Ehrig et al., 2005] Ehrig, M., Haase, P., Stojanovic, N., and Hefke, M. (2005). Similarity for ontologies - a comprehensive framework. In *13th European Conference on Information Systems.*

[Ehrig and Sure, 2004] Ehrig, M. and Sure, Y. (2004). Ontology mapping - an integrated approach. In *Proc. of the First European Semantic Web Symposium*, pages 76–91.

[Eklund et al., 2002] Eklund, P., Roberts, N., and Green, S. (2002). Ontorama: Browsing rdf ontologies using a hyperbolic-style browser. *Proc. of 1st Intl. Symposium on Cyber Worlds, 2002*, pages 405–411.

[Electronics and Telecommunications Research Institute] Electronics and Telecommunications Research Institute. ezOWL. `http://smi-protege.stanford.edu/svn/ezowl/`.

[Engels and Bremdal, 2001] Engels, R. and Bremdal, B. (2001). Corporum: a workbench for the semantic web. Semantic Web Mining Wokshop. PKDD/ECML.

[Euzenat, 1996a] Euzenat, J. (1996a). Corporate memory through cooperative creation of knowledge bases and hyper-documents. In *Proc. of the 10th Knowledge Acquisition, Modeling and Management for Knowledge-based Systems Workshop (KAW'96)*, pages 1–18.

[Euzenat, 1996b] Euzenat, J. (1996b). Hytropes: a www front-end to an object knowledge management system. In *Proc. of the 10th Knowledge Acquisition, Modeling and Management for Knowledge-based Systems Workshop (KAW'96)*, pages 1–12. Actes 10th knowledge acquisition workshop demonstration track.

[Euzénat et al., 2004] Euzénat, J., Loup, D., Touzani, M., and Valtchev, P. (2004). Ontology alignment with ola. In *Proc. of the 3rd EON Workshop at the 3rd Intl. Semantic Web Conference.*

[Falconer and Storey, 2008] Falconer, S. and Storey, M.-A. (2008). A cognitive support framework for ontology mapping. In *The Semantic Web – Proc. of the 6th Intl. Semantic Web Conf.*, pages 114–127.

[Fluit et al., 2004] Fluit, C., Sabou, M., and van Harmelen, F. (2004). Supporting user tasks through visualization of light-weight ontologies. In Staab, S. and Studer, R., editors, *Handbook on Ontologies*, pages 415–434. Springer.

[Fluit et al., 2005] Fluit, C., Sabou, M., and van Harmelen, F. (2005). Ontology-based information visualisation: Towards semantic web applications. In Geroimenko, V., editor, *Visualising the Semantic Web (2nd edition)*. Springer.

[Flynn] Flynn, J. VisioOWL. `http://mysite.verizon.net/jflynn12/VisioOWL/VisioOWL.htm` Last seen: 26.12.2008.

[FOAF] FOAF. Foaf – friend of a friend. `http://www.foaf-project.org/`.

[foafCORP] foafCORP. foafCORP. `http://rdfweb.org/foafcorp/intro.html`.

[FOAFnaut] FOAFnaut. FOAFnaut. `http://www.jibbering.com/foaf/`.

[Fuchs et al., 2003] Fuchs, M., Niederée, C., Hemmje, M., and Neuhold, E. J. (2003). Supporting model-based construction of semantic-enabled web applications. In *Proc. of Intl. Conf. on Web Information Systems Engineering*, pages 232–241, Los Alamitos, CA, USA. IEEE Computer Society.

[Furnas, 1986] Furnas, G. W. (1986). Generalized fisheye views. In *SIGCHI Bull. – Special issue: CHI '86 Conf. Proc.*, volume 17, pages 16–23, New York, NY, USA. ACM.

[Gansner and North, 2000] Gansner, E. R. and North, S. C. (2000). An open graph visualization system and its applications to software engineering. *Software — Practice and Experience*, 30(11):1203–1233.

[Gennusa, 2004] Gennusa, P. (2004). Ontopia's vizigator™– now you see it! In *Proc. of XML2004*, Washington, DC.

[Grosso et al., 1999] Grosso, E., Eriksson, H., Fergerson, R., Tu, S., and Musen, M. (1999). Knowledge-modeling at the millenium – the design and evolution of protégé -2000. In *Proc. of the 12th Intl. Workshop on Knowledge Acquisition, Modeling and Management (KAW'99)*.

[Horridge] Horridge, M. OWLVIz. `http://www.co-ode.org/downloads/owlviz/OWLVizGuide.pdf` Last seen 26.12.2008.

[Inselberg and Dimsdale, 1990] Inselberg, A. and Dimsdale, B. (1990). Parallel coordinates: A tool for visualizing multi-dimensional geometry. In *Proc. of the First IEEE Conference on Visualization (VIS '90)*, pages 361–378, Los Alamitos, CA, USA. IEEE Computer Society Press.

[Jung, 2005] Jung, J. J. (2005). Visualizing Recommendation Flow on Social Network. *Journal of Universal Computer Science*, 11(11):1780–1791.

[Jung, 2008a] Jung, J. J. (2008a). Query transformation based on semantic centrality in semantic social network. *Journal of Universal Computer Science*, 14(7):1031–1047.

[Jung, 2008b] Jung, J. J. (2008b). Ontology-based Context Synchronization for Ad-Hoc Social Collaborations. *Knowledge-Based Systems*, 21(7):573–580.

[Jung, 2009a] Jung, J. J. (2009a). Semantic Business Process Integration Based on Ontology Alignment. *Expert Systems with Applications*, 36(8):11013–11020.

[Jung, 2009b] Jung, J. J. (2009b). Knowledge Distribution via Shared Context between Blog-based Knowledge Management Systems: a Case Study of Collaborative Tagging. *Expert Systems with Applications*, 36(7):10627–10633.

[Keim et al., 1995] Keim, D., Kriegel, H.-P., and Ankerst, M. (1995). Recursive pattern: a technique for visualizing very large amounts of data. *Proc. of the IEEE Conf. on Visualization '95*, pages 279–286.

[Keim, 1996] Keim, D. A. (1996). Pixel-oriented visualization techniques for exploring very large databases. *Journal of Computational and Graphical Statistics*, 5:58–77.

[Kolonin] Kolonin, A. Webstructor. `http://www.webstructor.net`

[Lamping et al., 1995] Lamping, J., Rao, R., and Pirolli, P. (1995). A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In *Proc. of* *the SIGCHI Conf. on Human factors in computing systems (CHI '95)*, pages 401–408, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.

[Lanzenberger et al., 2003] Lanzenberger, M., Miksch, S., and Pohl, M. (2003). The stardinates – visualizing highly structured data. In *Proc. of the Intl. Conf. on Information Visualization (IV03)*, pages 47–52. IEEE Computer Society Press.

[Lanzenberger and Sampson, 2006] Lanzenberger, M. and Sampson, J. (2006). Alviz - a tool for visual ontology alignment. In *Proc. of Intl. Symposium of Visualization of the Semantic Web, (IV06-VSW)10th Intl. Conference Information Visualization*. IEEE Computer Society.

[Lenat and Guha, 1989] Lenat, D. and Guha, R. V. (1989). *Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[Martin and Eklund, 1999] Martin, P. and Eklund, P. (1999). Embedding knowledge in web documents. *Computer Networks*, 31(11-16):1403–1419.

[Mika, 2005] Mika, P. (2005). Flink: Semantic web technology for the extraction and analysis of social networks. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3(2-3):211–223.

[Milgram, 1967] Milgram, S. (1967). The small world problem. *Psychology Today*, 2:60–67.

[Miller et al., 1990] Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. (1990). Wordnet: An on-line lexical database. *Intl. Journal of Lexicography*, 3:235–244.

[Naeve, 2001] Naeve, A. (2001). The concept browser: a new form of knowledge management tool. In *Proc. of the 2nd European Web-based Learning Environments Conference (WBLE)*.

[Newman, 2004] Newman, M. E. J. (2004). Fast algorithm for detecting community structure in networks. *Physical Review E*, 69:066133.

[Ng, 2000] Ng, G. K.-C. (2000). *Interactive visualisation techniques for ontology development*. PhD thesis, University of Manchester, Manchester, UK.

[North and Shneiderman, 1997] North, C. and Shneiderman, B. (1997). A taxonomy of multiple window coordinations. Technical Report CS-TR-3854. citeseer.nj.nec.com/north97taxonomy.html.

[Noy, 2004] Noy, N. F. (2004). Ontology management with the prompt plugin. In *Proc. of the 7th Intl. Protégé Conf.*, Bethesda, Maryland.

[OntoViz] OntoViz. OntoViz: Visualizing protégé ontologies. `http://protegewiki.stanford.edu/index.php/OntoViz`.

[Pickett and Grinstein, 1988] Pickett, R. and Grinstein, G. (1988). Iconographic displays for visualizing multidimensional data. In *Proc. of the 1988 IEEE Intl. Conf. on Systems, Man, and Cybernetics*, volume 1, pages 514–519.

[Pietriga] Pietriga, E. IsaViz. `http://www.w3.org/2001/11/IsaViz`

[Pirolli et al., 2001] Pirolli, P., Card, S. K., and Wege, M. M. V. D. (2001). Visual information foraging in a focus + context visualization. In *Proc. of the SIGCHI Conf. on Human factors in computing systems (CHI '01)*, pages 506–513, New York, NY, USA. ACM.

[Pretorius, 2004] Pretorius, A. (2004). Lexon visualization: visualizing binary fact types in ontology bases. *Proc. of Information Visualisation, 2004 (IV 2004)*, pages 58–63.

[PromptViz] PromptViz. PromptViz. `http://www.thechiselgroup.org/promptviz`.

[Rector et al., 1997] Rector, A., Bechhofer, S., Goble, C., Horrocks, I., Nowlan, W., and Solomon, W. (1997). The grail concept modelling language for medical terminology. *Artificial Intelligence in Medicine*, 9:139–171.

[Robertson et al., 1991] Robertson, G. G., Mackinlay, J. D., and Card, S. K. (1991). Cone trees: animated 3d visualizations of hierarchical information. In *Proc. of the SIGCHI Conf. on Human factors in computing systems (CHI '91)*, pages 189–194, New York, NY, USA. ACM.

[Shahar and Cheng, 2000] Shahar, Y. and Cheng, C. (2000). Model-based visualization of temporal abstractions. *Computational Intelligence*, 16(2):279–306.

[Sintek] Sintek, M. The FRODO RDFSViz tool. `http://www.dfki.uni-kl.de/frodo/RDFSViz`

[Skuce and Lethbridge, 1995] Skuce, D. and Lethbridge, T. C. (1995). Code4: a unified system for managing conceptual knowledge. *Int. J. Hum.-Comput. Stud.*, 42(4):413–451.

[SRI International] International, S. The generic knowledge-base editor. `http://www.ai.sri.com/~gkb`

[Steer] Steer, D. RDFAuthor. `http://rdfweb.org/people/damian/RDFAuthor`

[Storey et al., 2001] Storey, M., Musen, M., Silva, J., Best, C., Ernst, N., Fergerson, R., and Noy, N. (2001). Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in protégé. In *Proc. of the Intl. Workshop on Interactive Tools for Knowledge Capture (K-CAP-2001)*.

[Storey and Muller, 1995] Storey, M.-A. D. and Muller, H. A. (1995). Manipulating and documenting software structures using shrimp views. In *Proc. of the Intl. Conf. on Software Maintenance (ICSM '95)*, pages 275–284, Washington, DC, USA. IEEE Computer Society.

[Stuckenschmidt et al., 2004] Stuckenschmidt, H., van Harmelen, F., de Waard, A., Scerri, T., Bhogal, R., van Buel, J., Crowlesmith, I., Fluit, C., Kampman, A., Broekstra, J., and van Mulligen, E. (2004). Exploring large document repositories with rdf technology: the dope project. *IEEE Intelligent Systems*, 19(3):34–40.

[Stumme and Maedche, 2001] Stumme, G. and Maedche, A. (2001). FCA-Merge: Bottom up merging of ontologies. In *Proc. of the 7th Intl. Conference on Artificial Intelligence*, pages 225–230.

[Telea et al., 2003] Telea, A., Frasincar, F., and Houben, G.-J. (2003). Visualisation of rdf(s)-based information. *Proc. of 7th Intl. Conf. on Information Visualization (IV 2003)*, pages 294–299.

[Touchgraph] Touchgraph. Touchgraph. `http://www.touchgraph.com`.

[Travers, 1989] Travers, M. (1989). A visual representation for knowledge structures. In *Proc. of the 2nd ACM Conf. on Hypertext*, pages 147–158, New York, NY, USA. ACM.

[Tufte, 2001] Tufte, E. (2001). *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, CT, 2nd edition edition.

[Tzitzikas et al., 2007] Tzitzikas, Y., Kotzinos, D., and Theoharis, Y. (2007). On Ranking RDF Schema Elements (and its Application in Visualization). *Journal of Universal Computer Science*, 13(12):1854–1880.

[van Ham and van Wijk, 2004] van Ham, F. and van Wijk, J. (2004). Interactive visualization of small world graphs. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 199–206. CSPress.

[Wertheimer, 1967] Wertheimer, M. (1967). Laws of organization in perceptual forms. In *A source book of Gestalt psychology*. Humanities Press, New York.

[Whiting and Cramer, 2002] Whiting, M. A. and Cramer, N. (2002). Webtheme$^{\mathrm{TM}}$: Understanding web information through visual analytics. In *Proc. of the 1st Intl. Semantic Web Conf (ISWC '02)*, number 2342 in LNCS, pages 460–468, London, UK. Springer-Verlag.

[Wienhofen, 2004] Wienhofen, L. (2004). Using graphically represented ontologies for searching content on the semantic web. *Proc.of the 8th Intl. Conf. on Information Visualisation (IV 2004)*, pages 801–806.

[Wise et al., 1995] Wise, J., Thomas, J., Pennock, K., Lantrip, D., Pottier, M., Schur, A., and Crow, V. (1995). Visualizing the non-visual: spatial analysis and interaction with information from text documents. In *Proc. of IEEE Symposium on Information Visualization*, volume 0, page 51, Los Alamitos, CA, USA. IEEE Computer Society.

[Ziegler et al., 2002] Ziegler, J., Kunz, C., and Botsch, V. (2002). Matrix browser: visualizing and exploring large networked information spaces. In *CHI '02: extended* *abstracts on Human factors in computing systems*, pages 602–603, New York, NY, USA. ACM.