The SemSearchXplorer – Exploring Semantic Search Results with Semantic Visualizations

Thomas Daniel Ullmann, Victoria Uren[*], Andriy Nikolov

Knowledge Media Institute (KMi)
The Open University
Milton Keynes, United Kingdom
{t.ullmann, a.nikolov}@open.ac.uk
v.uren@dcs.shef.ac.uk

Abstract: SemSearchXplorer is a toolkit for the exploration of semantic data. The goal is to lower user barriers to access information in semantic data repositories. Therefore SemSearchXplorer supports the user in three respects: (1) it supports querying of the semantic data with a keyword based approach, so the users do not need to learn a semantic query language, (2) it helps users find relevant results both by using semantic enriched information about the results and semantic filter options to narrow down the set of results, and (3) it provides information exploration capabilities through semantic visualizations recommended by the system. Filtering of semantic search results helps to narrow down the result set to a more manageable amount of information. Besides searching for relevant information, facilities for the exploration of the results help users to gain insight in the context of results. With several semantic visualizations, we try to help users making sense of the raw data. Based on the assumption that there is no single visualization that fits all exploration needs, SemSearchXplorer recommends visualizations based on the selected information of users.

Keywords: semantic search, semantic visualization, information exploration.

---

[*] Present address: Department of Computer Science, Regent Court, 211 Portobello, University of Sheffield, Sheffield, S1 4DP United Kingdom.

1 From SemSearch to SemSearchXplorer

Semantic search engines return semantic enriched results of backing data structure, which appropriate visualized support the searching and exploration tasks of knowledge workers. We can subsume two strands of research associated with semantic search. Searching for ontologies (e.g. Watson [Aq08] or Swoogle [Di04]) and searching within semantic data. The focus of this work lies on the latter one. With SemSearchXplorer, we extended the semantic search engine SemSearch [LUM06] to a toolkit for exploring semantic data following the notion of the importance of iterative and exploratory search modes to the usability of search systems [Ur07].

With the keyword-based query interface of SemSearch, users do not need to learn semantic query languages, like SPARQL [PS08] or SeRQL[1], to query semantic data. SemSearch translates automatically users' keywords into semantic queries. The metaphor for searching semantic knowledge-spaces with SemSearch is similar to keyword-based search engines, and thus familiar for users. SemSearch returns, as query results, instances and related triples (subject-predicate-object). In contrast to query results not based on semantic web technology, these triples contain structured information reflecting the relational information of the repository. Every result contains a view to the backing data. The advantage of semantic data over unstructured data is that this result can serve as a starting point for the exploration of the context of the information due to the interconnected nature of ontologies.

SemSearchXplorer makes use of this structured information. Its user interface is based on the knowledge lenses metaphor [Ur08]. This means that it provides the users with several lenses or views to the data repository. The type of lens used is dependent on information selected by users to explore the context of the SemSearch query result.

A common approach to visualize ontologies is an indented list [Ka07]; this is a tree view like visualization of the ontology, for example used in the Neon Toolkit [Ha08] or Protégé [No01]. Indented lists are simple to implement and users are familiar with this concept, because of its common use in web-search engines. The downside of list visualizations is that they can only represent tree and not graph structures. Graph structures would be more suitable to visualize the interconnected structure of semantic search results. However, it is not only list visualizations that have certain strengths and weaknesses. There is no single visualization that fits every user information exploration need [Ka07]. With SemSearchXplorer, we provide a range of visualization types for the backing data with the goal of providing appropriate lenses or views based on content.

---

[1] http://www.openrdf.org/doc/sesame/users/ch06.html

We made the design decision for a content-based recommendation because of the requirement that the search engine must be able to work out-of-the-box. A solution based on personal or collaborative recommendations would have the benefit of considering user characteristics, but would have the drawback that such systems only works after the system has gathered some information about the user (cold start problem).

While SemSearch supports users with a keyword-based query interface, SemSearchXplorer focuses on the exploration of the retrieved results. We support this exploration process with several semantic visualizations. Semantic visualization, or ontology-based visualization [FSH03], is part of information visualization [CMS99, Wa00, Sp07] and focuses on visualizing classes, instances, properties, and their multiple relations. It is a part of the top layer of the semantic web stack [Be00] - the user interface and application layer. The aim of semantic visualization is to visualize the underlying structure of semantic data. One vision of semantic visualization would be to use the content characteristics of the semantic web specification for the automatic creation of visualizations.

2 Overview of SemSearchXplorer

One of the primary goals of SemSearchXplorer is to support the users with their knowledge working tasks. Therefore, we defined two general requirements:

- Hide technical details of semantic technology and semantic visualizations techniques from users

- Enhance the information search and exploration experience for users.

To meet the first requirement SemSearchXplorer uses SemSearch, which allows using a google-like keyword search. Users have no need to learn semantic query languages to query for information of the semantic web. For the details of SemSearch see [LUM06].

For the second requirement, SemSearchXplorer adds on top of the layer architecture of SemSearch a visualization layer (see fig. 1). This layer is built as a pipeline. The query results of SemSearch are the starting point of the pipeline. Based on user interaction with SemSearchXplorer the filter and recommendation units set the course for either the amount of processed information (filtering) or the type of the presented visualization (recommendation). According to the recommended visualization and the filter options, the engine transforms query results into the data model for the visualization (e.g. table, graph or tree data structures). Then we enrich the data model with visualisation information (e.g. colour, size or layout information) to the visual form. The last unit of the pipeline is the view. A graphical renderer plots the actual view on the screen.

Based on the actions of the users, either a new query is sent to SemSearch, or if a visualization does not need new information from the search engine, the SemSearch query results are transformed in another data model on which the new view is based.
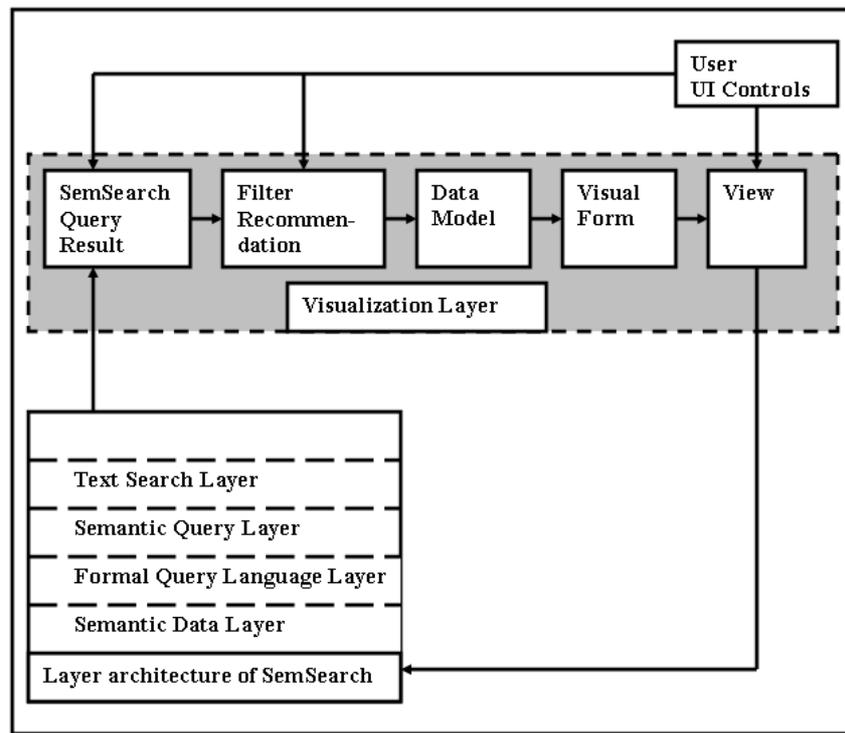

Figure 1: Layer architecture of SemSearchXplorer

SemSearchXplorer uses this architecture to provide knowledge workers with automatically recommended visualizations to support them with their information exploration activity.

SemSearchXplorer is built as a Java Swing application. It is meant to be used as a browser application using the Java Web Start Technology.

3 Supporting Information Exploration with SemSearchXplorer

The Semantic Web is a web of connected resources. Resources are connected through predicates with other resources forming a graph of resources. This concept allows users to start their information exploration process with a resource and then explore the context, the related resources of the chosen starting resource. It is important for the user to find the right starting point for his search. Once found, the user can explore the context of the information quite easily due to the underlying semantic web technology.

In many cases, the user does not now much about the information in the knowledge base. For this case, it is convenient for the knowledge worker to type in keywords into a search interface to view the results the query engine returns. SemSearchXplorer uses the semantic search engine SemSearch, which provides a keyword based query interface. Users start their information exploration process with keyword queries. SemSearch transforms these keywords into semantic queries and returns a ranked result set of the query. Based on this result set SemSearchXplorer supports the user with different lenses or visualizations of the result set helping him to refine and explore the information.

3.1 Searching

The query interface of SemSearchXplorer (see fig. 2) allows either to search for a keyword or to specify the expected type (the subject) of search results. The later is supported through the syntax "subject:keyword". With Boolean operators like and/or we can compose complex queries (for details see [LUM06]). For example a query about news of projects has the following form: "news:project".

Figure 2: Initial query interface of SemSearchXplorer

3.2 Refining the Result Set

SemSearchXplorer visualizes the query results as an ordered list (see fig. 3). A list representation has the advantage that users can read very quickly through the text and scan for information. The information needs little space to plot. The order of the set of results is sorted according to the ranking mechanism of SemSearch. Each hit contains the instance using the fragment part of an URI, or if available, the label information (rdfs:label). In addition to the instance we provide context information consisting of the class names and the relations to other instances, which belong to the query. This makes it easier for users to search for relevant hits. Each instance also provides a tooltip with statistical information about the result.
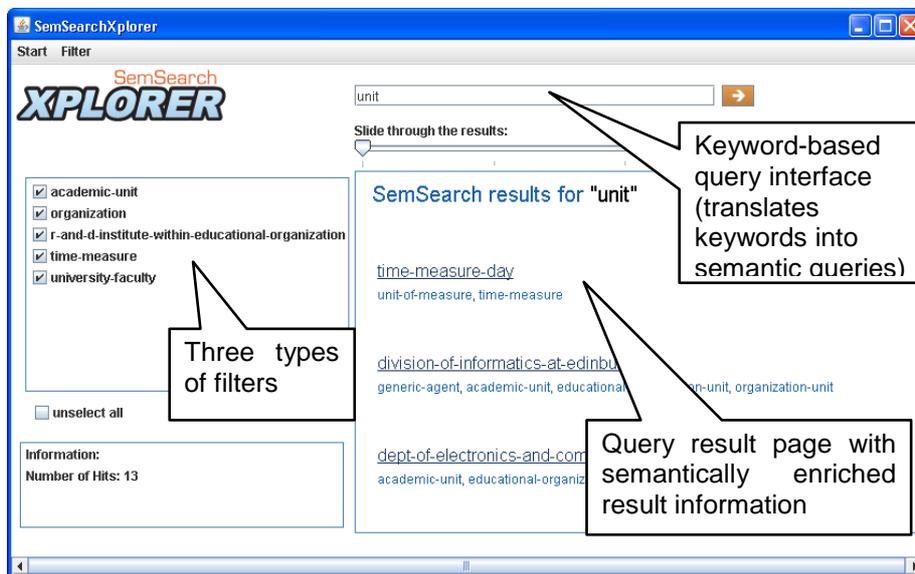

Figure 3: SemSearchXplorer query result page

For users is it important to be able to narrow down the result set to a set of interesting results. There are tradeoffs between filters with a small number of general class items, which allow a quick reduction of the result set, and filters with a large number of filter items, which allow a fine-grained narrowing, but have the drawback that the user has to select from a large set of filter possibilities. To allow filtering of the result set, we implement three types of filters. The first filter uses the direct super-classes of an instance but not all the other classes higher in the class hierarchy. Compared to the second filter, the all class filter, which uses the whole class information of the class hierarchy of the result set, the direct class filter generates less filter items, while the all class filter produces more filter items allowing precise filtering. The third filter uses the key concepts (KCE – Key Concept Extraction [PMA08]) of an ontology. With KCE it is possible to extract the 'best descriptors' of an ontology. Therefore, the algorithm tries to find the most important concepts of an ontology as a human expert would do. By now, the KCE algorithm considers three concepts. Natural categories (that are concepts that are information-rich in psycholinguistic sense), density (concepts which are information-rich in an ontological sense), and popularity (frequency of terms returned from a query to the Yahoo search engine). Although this algorithm's goal is to generate automatic ontology summaries, we consider KCE as a method applicable for filtering the result set according to the most meaningful items of an ontology while ignoring classes that are less important. Although the algorithm tries to maximize the coverage of the ontology, some concepts are not contained in the filter. We add hits of classes not belonging to the key concepts or subclasses to a new filter entry called "other".

With these three types of filters, the user has effective mechanism to filter quickly to relevant information. Once found, the user can start the exploration of the hit.

3.3 Exploration

After the user selected a result, a new view of SemSearchXplorer shows the exploration page (see fig. 4). By now, we support users with three types of visualizations to explore the context of the selected hit: a graph, cluster and chart visualization.
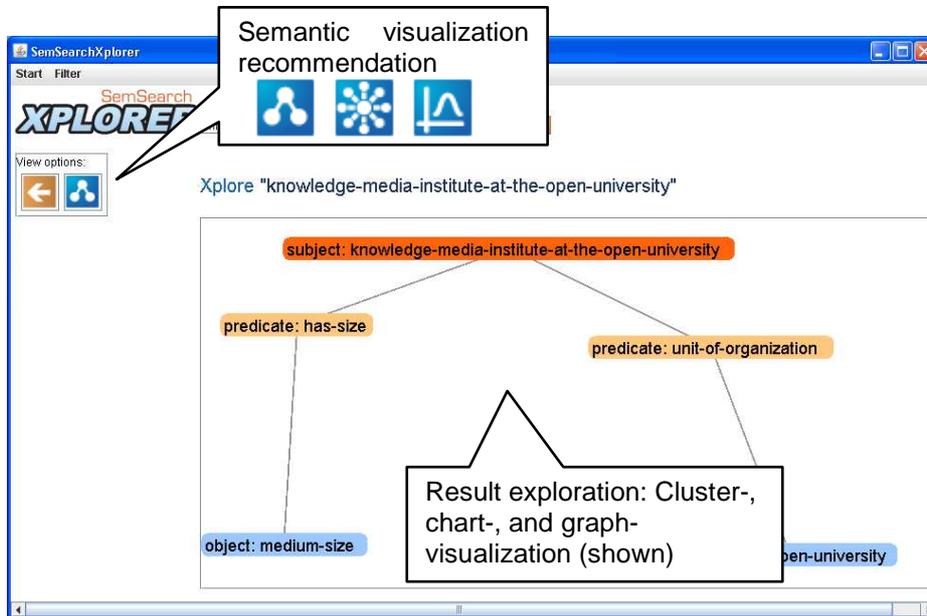
Figure 4: SemSearchXplorer exploration page with graph visualization

Graph visualizations are suitable for representing the context of a result. In contrast to a list-based visualization, a graph visualization is able to represent the connections between classes and instances. The list representation does not reflect the relational structure of the semantic result. Within our graph visualization instances are connected with edges representing the properties of the semantic result. While graph visualization helps users to understand the relations between resources, large graph visualizations become cluttered or need a large space on the screen.

Our cluster visualization (see fig. 5) is used for showing larger amounts of information. Information that is not necessary for the first exploration is hidden in a cluster. If users need this information, they can expand the cluster. With this technique, we can reduce the problem of cluttered visualizations to a certain degree. As the graph visualization the cluster visualization tries to support users making sense of the relational structure of the backing data.
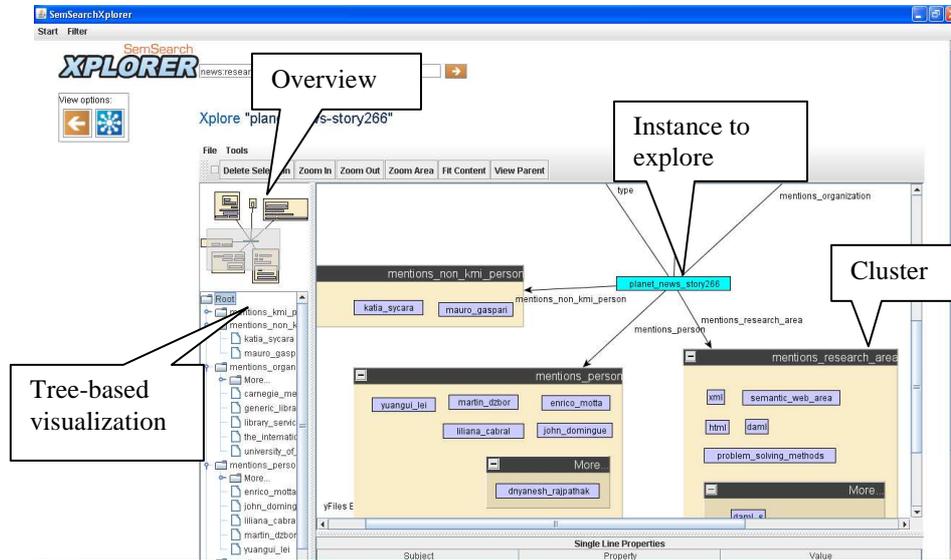
Figure 5: SemSearchXplorer exploration page with cluster visualization

Features of the cluster visualization are:

- The instance to explore is highlighted and central in the graph representation.

- The object instances of the triple are ordered in a cluster with the name of the property. A cluster can contain a "more" cluster. We order higher ranked instances into the first level group and lower ranked instances into the "more" cluster.

- On the top left position of the cluster visualization we provide an overview and below the overview a tree representation of the whole visualization.

However, not only could the information about the context be interesting for users but also statistical information about the instance. This allows users to compare instance according their relevance. For an example of our chart visualization (see fig. 6). The outer right blue chart of the chart visualization has the meaning that 12 instances have as line manger Enrico Motta.
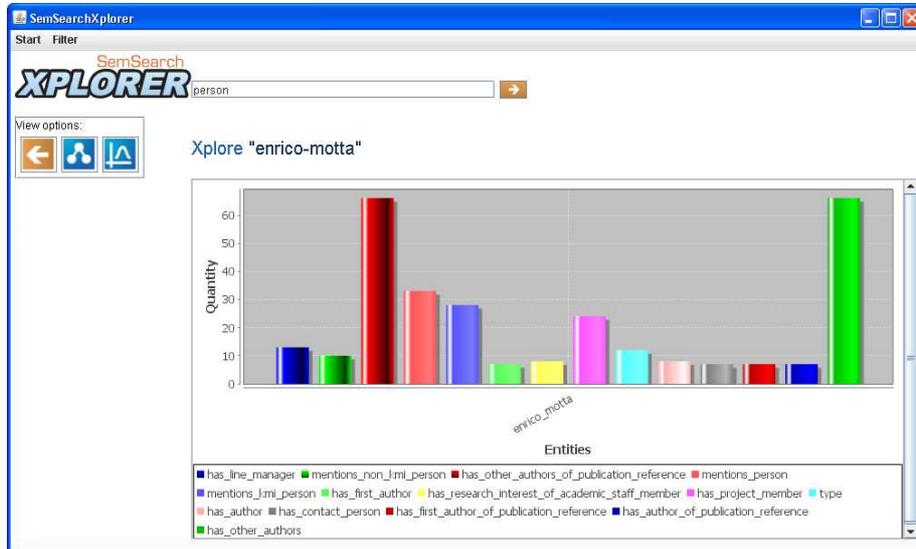
Figure 6: SemSearchXplorer exploration page with chart visualization

For a detailed discussion of the cluster and chart visualization and the underlying algorithms, see [FUN09].

3.4 Recommendation

The most notable feature of the exploration page is the recommendation component consisting of a variable number of recommended visualizations. Until now, we have focused on content-based recommendations of visualizations. The goal would be to provide users the most suitable view for a particular set of data (consisting of instances, classes and/or relations). This is quite an ambitious vision, but for the first implementation of SemSearchXplorer we consider heuristics which serve as starting point for future recommendations. The heuristics follow these simple rules:

- If the number of properties and related instances is above a threshold, then offer a cluster visualization.

- If this number is below a threshold, then offer a graph visualization.

- If filtering influences this number significantly, then provide cluster or graph visualization according to the threshold.

- For chart visualizations, we need three clusters of data (x and y-axis, and the values to plot). Therefore, if we can generate three clusters of data, then offer a chart visualization.

Based on the chosen entities of the user SemSearchXplorer recommends the user visualizations, which are able to visualize the selected information. As seen in figure 4 the recommendation of visualization is represented in iconic form below the SemSearchXplorer logo. From left to right the icons are ordered according to which of the possible visualizations the system predicts are more suitable.

4 Further prospects

The goal of SemSearchXplorer is to help people find relevant information more quickly and to help them to explore the context of this information. For the case that users do not exactly know what they are looking for, semantic search (SemSearch) supports the user in finding a suitable starting point for their exploration and with the visualization layer of SemSearchXplorer semantic visualizations help to explore the context of the result.

SemSearchXplorer recommends visualizations according to the information the user provides and the capability of the visualizations to make use of the provided information. Thus, the decision is more a technical one, than based on user experience. Further research will examine what questions the users want to answer with SemSearchXplorer and what visualizations will help them doing that.

Another aspect concerns questions of user trust in information. The question is, do people trust the visualized results more, if they can reproduce how the results are generated and what sources of data are involved in the process. An explanation component could help to unveil the mechanisms of the result generation. This component provides more information than the actual info box of SemSearchXplorer, which only shows basic statistical information of the query results.

After the user entered the query, a list-based representation of the visualization is generated. The advantage of lists is that users are accustomed to read quickly through linear text. This representation works well, if the user is only interested in the highest ranked query results. However, if the user wants to get an overview of the whole result set another visualization, which provides an overview of all hits at once, would be more suitable.

With SemSearchXplorer and its software architecture we have an extensible testbed to examine these outlined user centred and technical questions.

Acknowledgements

References

[Aq08]   d'Aquin, M.; Sabou, M.; Motta, E.; Angeletou, S.; Gridinoc, L.; Lopez, V.; Zablith, F.: What can be done with the Semantic Web? An Overview of Watson-based Applications, 5th Workshop on Semantic Web Applications and Perspectives, SWAP 2008, Rome, Italy, 2008.

[Be00]   Berners-Lee, T.: Semantic Web – XML2000. W3 Talks, http://www.w3.org/2000/Talks/1206-xml2k-tbl/, 2000.

[CMS99] Card, S. K.; Mackinlay, J. D.; Shneiderman, B.: Readings in information visualization: using vision to think. Morgan Kaufmann Publishers, San Francisco, 1999.

[Di04]   Ding, L.; Finin, T.; Joshi, A.; Pan, R.; Scott Cost, R.; Peng, Y.; Reddivari, P.; Doshi, V. C.; Sachs, J.: Swoogle: A Search and Metadata Engine for the Semantic Web. Proceedings of the 13th ACM Conference on Information and Knowledge Management, November, 2004.

[FSH03] Fluit, C.; Sabou, M.; Harmelen, F.: Ontology-based Information Visualization. In (Geroimenko, V.; Chen, C.; eds.): Visualizing the Semantic Web, pp 36-48.London: Springer, 2003.

[FUN09] Nazir, F.; Uren, V.; Nikolov, A.: Algorithms for Generating Ontology Based Visualizations from Semantic Search Results, accepted paper for the Workshop MoViX '09, 2009.

[Ha08]   Haase, P.; Lewen, H.; Studer, R.; Tran, T.; Erdmann, M.; d'Aquin, M.; Motta, E.: The NeOn Ontology Engineering Toolkit. WWW 2008, April 21-25, 2008, Beijing, China.

[Ka07]   Katifori, A; Halatsis, C; Lepouras, G.; Vassilakis, C.; Giannopoulou, E.: Ontology Visualization Methods – A survey. ACM Computing Survey, Vol. 39, No 4, October 2007; pp. 10.1-10.42.

[LUM06] Lei, Y.; Uren, V.; Motta, E.: SemSearch: A Search Engine for the Semantic Web. , In (Staab, S.; Svatek, V.; eds.): Managing Knowledge in a World of Networks, 15th International Conference, EKAW 2006, Podebrady, Czech Republic, October 2-6, 2006, Proceedings. Lecture Notes in Computer Science 4248 Springer 2006; pp. 238-245.

[No01]   Noy, N. F.; Sintek, M.; Decker, S.; Crubezy, M.; Fergerson, R. W.; Musen, M. A.: Creating Semantic Web Contents with Protégé-2000. IEEE Inteligent Systems, March/April 2001, pp. 60-71.

[PMA08] Peroni, S., Motta, E., d'Aquin, M.: Identifying key concepts in an ontology through the integration of cognitive principles with statistical and topological measures, Third Asian Semantic Web Conference, Bangkok, Thailand, 2008

[PS08]   Prud'hommeaux, E.; Seaborne, A.: SPARQL Query Language for RDF. W3C Recommendation. January, 2008. http://www.w3.org/TR/rdf-sparql-query/, [2009-04-21].

[Sp07]   Spence, R.: Information Visualization: Design for Interaction. Prentice Hall, 2007.

[Ur07]   Uren, V.;Lei, Y.; Lopez, V.; Liu, H.; Motta, E.; Giordanino, M.: The usability of semantic search tools: a review. The Knowledge Engineering Review, Vol. 22:4, 2007; pp 361-377.

[Ur08]   Uren, V.; Franz, T.; Cimino, P.; Lei, Y.; Dadzie, A.; Gilardoni, L.: X-Media Deliverable 4.4. Requirements for Knowledge Sharing Tools in Phase 2. Technical Report of the X-Media project, 2008.

[Wa00]   Ware, C.: Information Visualization: Perception for Design. Morgan Kaufmann Publishers Inc, 2000.