QuizRDF: Search Technology for the Semantic Web

John Davies and Richard Weeks
BT Exact,
Orion 5/12, Adastral Park,
Ipswich IP5 3RE, UK.
john.nj.davies@bt.com

Abstract

An information-seeking system is described which combines traditional keyword querying of WWW resources with the ability to browse and query against RDF annotations of those resources. RDF(S) and RDF are used to specify and populate an ontology and the resultant RDF annotations are then indexed along with the full text of the annotated resources. The resultant index allows both keyword querying against the full text of the document and the literal values occurring in the RDF annotations, along with the ability to browse and query the ontology. We motivate our approach as a key enabler for fully exploiting the Semantic Web in the area of knowledge management and argue that the ability to combine searching and browsing behaviours more fully supports a typical information-seeking task. The approach is characterised as "low threshold, high ceiling" in the sense that where RDF annotations exist they are exploited for an improved information-seeking experience but where they do not yet exist, a search capability is still available.

1. Introduction

There are now more than two billion documents in the WWW, which are used by more than 300 million users globally, and millions more pages on corporate intranets. The continued rapid growth in information volume makes it increasingly difficult to find, organise, access and maintain the information required by users. A semantic web has been proposed [1] that provides enhanced information access based on the exploitation of machine-processable meta data. We are particularly interested in the new possibilities afforded by semantic web technology in the area of knowledge management and we discuss this below before moving on in the rest of the paper to describe a specific example of a KM tool, the QuizRDF search engine.

Central to the vision of the Semantic Web are ontologies. Ontologies are seen as facilitating knowledge sharing and re-use between agents, be they human or artificial [2]. They offer this capability by providing a consensual and formal conceptualisation of a given domain. As such, the use of ontologies and supporting tools offer an opportunity to significantly improve knowledge management capabilities in large organisations and it is their use in this particular area which is the subject of this paper.

1.1 The Semantic Web and Knowledge Management

Due to a number of factors, including globalisation and the impact of the Internet, many organisations are increasingly geographically dispersed and organised around virtual teams. As noted in, for example, [3], such organisations need knowledge management and organisational memory tools that encourage users to understand each other's changing contextual knowledge and foster collaboration while capturing, representing and interpreting the knowledge resources of their organisations.

Important information is often scattered across Web and/or intranet resources. Traditional search engines return ranked retrieval lists that offer little or no information on the semantic relationships among documents. Knowledge workers spend a substantial amount of their time browsing and reading to find out how documents are related to one another and where each falls into the overall structure of the problem domain. Yet only when knowledge workers begin to locate the similarities and differences among pieces of information do they move into an essential part of their work: building relationships to create new knowledge.

Information retrieval traditionally focuses on the relationship between a given query and the information store. On the other hand, exploitation of interrelationships between selected pieces of

information (which can be facilitated by the use of ontologies) can put otherwise isolated information into a meaningful context. The implicit structures so revealed help users use and manage information more efficiently [4].

Ontologies offer a way to deal with heterogeneous representations of Web resources and their interrelationships. The domain model implicit in an ontology can be taken as a unifying structure for giving information a common representation and semantics. Provided with an ontology meeting the needs of a particular user community, knowledge management tools can arrange knowledge assets into the predefined conceptual classes of the ontology, allowing more natural and intuitive access to knowledge. QuizRDF is an example of such a tool.

1.2 Search Technology for the Semantic Web

In this subsection, we motivate the design of QuizRDF, a search engine that uniquely combines free-text search with a capability to exploit RDF metadata in searching and browsing. There are 3 primary reasons for this approach, 2 of which are based on theoretical observations, one of which is more pragmatic.

Pragmatically speaking, it is the case at the time of writing that only a very small proportion of WWW- and intranet-based information resources are annotated with RDF (meta)data. It is therefore preferable to provide a combined search facility that can exploit metadata annotations where they exist but which will degrade gracefully to a "traditional" free text search engine where information is not annotated.

Turning to more principled reasons for our approach, data from the Information Retrieval literature indicates that information seeking activity is often comprised of a mixture of searching and browsing behaviours. Research on user behaviour has typically characterised a variety of paradigms for information seeking and [5], for example, provides a discussion of these paradigms and their interactions. Similarly, [6], [12] and [13], for example, describe systems for combining browsing and searching of WWW resources and argue that this combination delivers a more powerful tool for information seeking than search or browse facilities alone. It is this blended querying and browsing approach that QuizRDF adopts in the context of RDF(S)-based ontologies.

More specifically to the discussion at hand, browsing a graphical display of a comprehensive ontology can result in a high cognitive overload for the user. Instead of navigating the entire graph of a complex ontology, users may benefit more by starting from a particular node of the ontology and exploring its immediate surroundings in order to create appropriate queries. This raises the question of how to enable the user to find an interesting node in the ontology from which to start his exploration. In QuizRDF, as we will see, this is achieved by user entry of a "standard" keyword search query which is used to locate them at an appropriate point in the information space represented by the ontology.

Another important observation is that it is in the general case impossible and impractical to cover the content of a document exhaustively by an RDF description. In practice, RDF descriptions can never replace the original document's content: any given RDF description of a set of resources will inevitably give one particular perspective on the information described. Essentially, a metadata description can never be complete since all possible uses for or perspectives on data can never enumerated in advance. Searches restricted to RDF descriptions will tend to produce a lower recall, while it is especially important at the beginning of a retrieval session to achieve a high recall. Most users are not able to initiate their search by formulating a complex and precise query. They prefer to start with a very simple query consisting of only one or two search terms in order to get a first idea of what information is available. Users may then continue their search by refining the queries to narrow the search results down to relevant documents.

Searching the full text of documents along with any associated RDF annotations can ensure the high recall desirable in early stages of the information seeking process. In later stages of the search, when the user may typically be more interested in the precision of the retrieval results, it can be advantageous to put more emphasis on searching the RDF annotations.

QuizRDF can be used like a conventional Internet search engine by entering a set of search terms or a natural language query and produces a ranked list of links to relevant Web pages based on statistical algorithms [7] in the usual way. However, QuizRDF*'s* indexing and retrieval technique is also designed to use domain knowledge that is made available in the form of ontologies specified as RDF Schemas. In our data model, RDFS is used to specify the classes in the ontology and their properties. The information items processed by QuizRDF are then RDF resources, which may be Web pages or parts thereof. Ontologically speaking, these RDF resources (WWW pages or parts thereof) are thus instances of the classes defined in RDFS.

2. Ontological Indexing

QuizRDF*'s* indexing and retrieval technique is designed to use domain knowledge that is made

available in the form of ontologies specified as RDF Schemas. The information items processed by QuizRDF are generally RDF resources, which may be whole Web pages or parts of Web pages. QuizRDF uses a given RDF Schema to create a structured index of RDF resources.

The core of the indexing process in QuizRDF is the assignation of content descriptors to RDF resources (Web pages or parts thereof). Content descriptors of a resource are terms (words and phrases) that QuizRDF obtains from both a full text analysis of the resource content *and* from processing all literal values that are directly related to the resource by a property (recall that an RDF resource is an instance of a class in the ontology). The QuizRDF index also retains structural information about the ontology from the corresponding RDF(S) description (e.g. classes, their properties and the sub/superclass relations holding between them).

In practice, the ontological index created by QuizRDF is a set of triples that refer to a set of RDF resources in a manner analogous to the way in which constituent terms (words and phrases) refer to documents in "traditional" information retrieval systems [10].

In the RDF metadata, URLs are instances of classes, as defined by the type-of property and we can write this in our scheme as:

$$<URL_n, \text{type-of}, \text{Employee}>$$

Furthermore, values of properties can be written down as follows:

$$<URL_n, \text{last\_name}, \text{"Miller"}>$$

QuizRDF creates a multidimensional index by combining such triples as follows:

$$<\text{"Miller"}, \text{Employee}, \text{last\_name}> \rightarrow URL_n$$

which represents the fact that the resource at URLn is an instance of class employee and that this instance's lastname property has value "Miller". More generally, a set of triples of the following type are produced:

$$<\text{literal,class,property}> \rightarrow \text{URL}$$

At the same time the full text of the annotated URLs is indexed in the way familiar from the information retrieval literature , (conceptually at least) creating further triples:

$$<\text{"George Miller"}, \text{Employee}, \phi> \rightarrow URL_n$$

So the triple above represents the fact that the phrase "George Miller" occurs in the body of the document at URL_n and that this document is of type *Employee*.

Figure 1 shows a simple example to illustrate ontology-based indexing. When indexing the example Web page at *malta.bt.com/gm/cv*, QuizRDF not only analyses the full text of the resource content but also the relevant parts of the RDF graph that describe this resource.

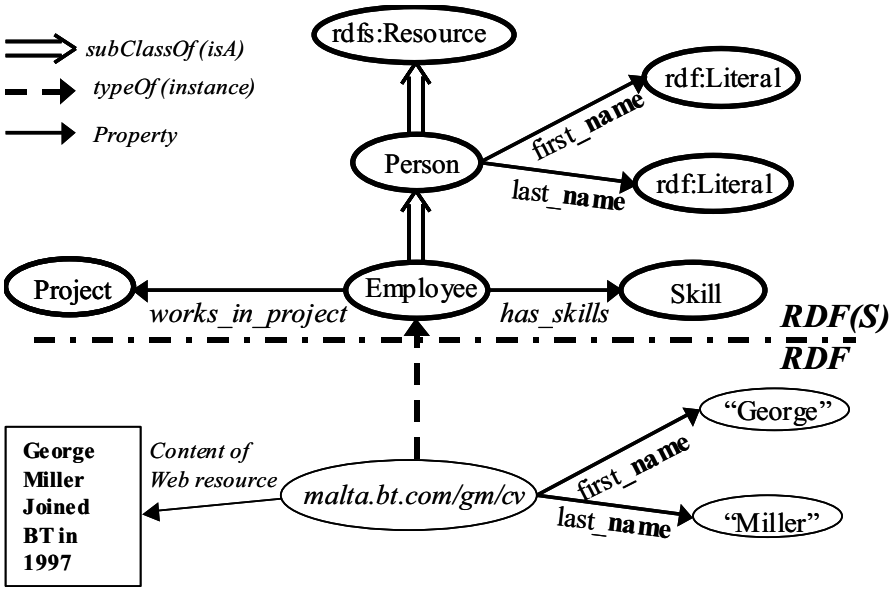

Figure 1. Ontology-based indexing.

Since our example Web page is annotated as being of type *Employee*, QuizRDF processes all literal values

that are directly related to the class *Employee*. The properties connected to *Employee* in this example are *last_name*, *first_name*, *has_skills* and *works_in_project* (*last_name* and *first_name* being inherited from the class *Person)*. QuizRDF traverses the RDF graph along these properties and performs a full text analysis on those properties having literal values. (Note that QuizRDF will also convert numeric literal values to strings for similar indexing). The resulting index contains descriptors extracted from both the full text content of the Web page and the RDF graph.

The obtained content descriptors are stored in QuizRDF's index along with references to their structural origin. Figure 2 illustrates schematically the data structure used to store the index for the given example. For each extracted keyword the index maintains a reference to the resource type Employee. For descriptors obtained from literal values, the database additionally stores the name of property that relates the literal to the given instance of *Employee.*

Descriptor	Class	Property	Resource
Miller	Employee	Ø	malta.bt.com/gm/cv
joined	Employee	Ø	malta.bt.com/gm/cv
BT	Employee	Ø	malta.bt.com/gm/cv
1990	Employee	Ø	malta.bt.com/gm/cv
George	Employee	first_name	malta.bt.com/gm/cv
Miller	Employee	last_name	malta.bt.com/gm/cv

Figure 2. Ontology-based index

Figure 3 below gives a more general overview of the full QuizRDF index structure.

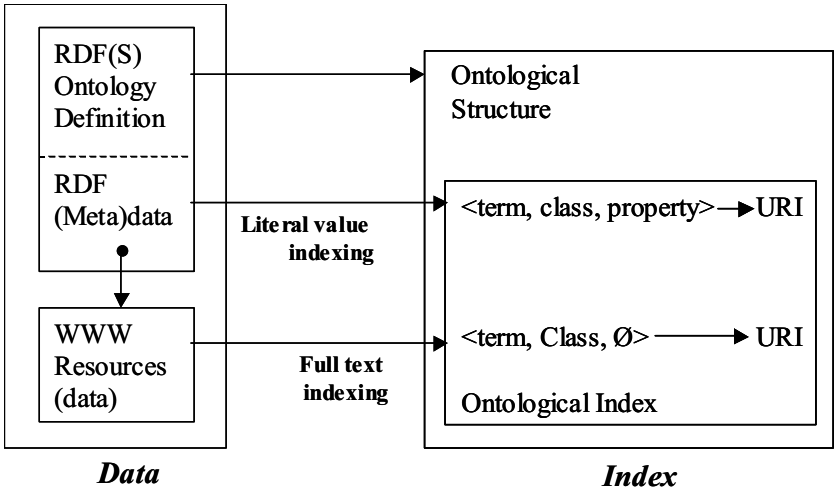

Figure 3. Ontological Index Structure

3. Ontological Searching

In this section, we describe how the ontological index above is used to provide a natural and intuitive browsing and searching interface onto a set of RDF-annotated WWW information resources.

On start-up QuizRDF presents the user with a text entry box and a drop-down menu. The drop-down menu contains a list of all the resource types stored in the QuizRDF index. The user can enter any natural language text into the text entry box. QuizRDF responds by returning a list of RDF resources ranked according to a resource's relevance to the user query (the ranking is currently based on a variation of the well-known *tf.idf* vector product scheme [11]). Simultaneously, the classes of which the URLs in the results list are instances are computed and included in the drop-down list above the results list. Selecting a class then (i) filters the retrieval list to include only those URLs which are instances of the selected class and (ii) displays the properties and related classes to the selected class, each of which has a hyperlink associated with it allowing the user to browse the ontology.

For each attribute the user can input a search criterion. QuizRDF combines the search criteria entered (which can be both free text search terms and attribute values) and matches the resulting query against its ontology-based index. In addition, resource types (classes) related by some property to the currently selected type are displayed as hyperlinks. Clicking on such a type then selects that type and in turn displays those types which are related to it. Thus the user can browse the ontology in a natural and intuitive way.

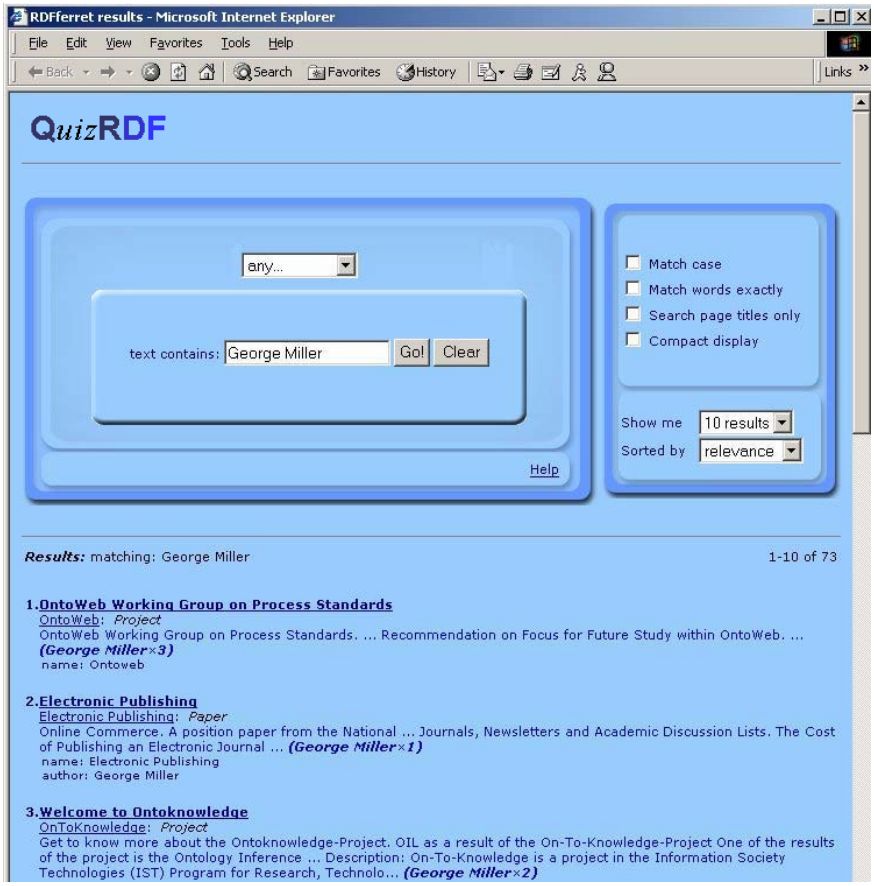

Figure 4.

To exemplify, Figure 4 shows a typical initial query by a user using the skills ontology described above. The user has entered a free text query for information about an employee called George Miller. The search engine has returned a ranked list of 73 documents mentioning the terms "George" and/or "Miller". At the top of the screenshot can be seen a drop-down list containing the selection "any…". When returning the 73 results documents, QuizRDF has also compiled a list of the classes to which each document belongs. This class list is then made available to the user via the drop-down list referred to.

Figure 5 shows the result after the user has selected the *Employee* class from the drop-down list. The screen now shows the properties of the Employee class, differentiating between those which relate (instances of) this class to (instances of) another class (*HasSkills* and *WorksInProject*) and those which expect a literal

value (*last name* and *first name*). The user has then specified values (Miller and George respectively) for these properties, while now leaving the free text search box empty. Based on these selections, QuizRDF has identified in its results list the single document (instance) of class *Employee* fulfilling the criteria specified. The superclasses of Employee (*Person* and *Resource*) are also identified. These superclasses, as well as the classes linked to *Employee* by properties (*Skills* and *Projects*), are clickable, allowing the user to continue to browse the ontology. While browsing the ontology, the user can also enter search text at any point, with the free-text search thus performed being restricted to the current class if one is selected. In this way, seamless switching between querying and browsing is supported.

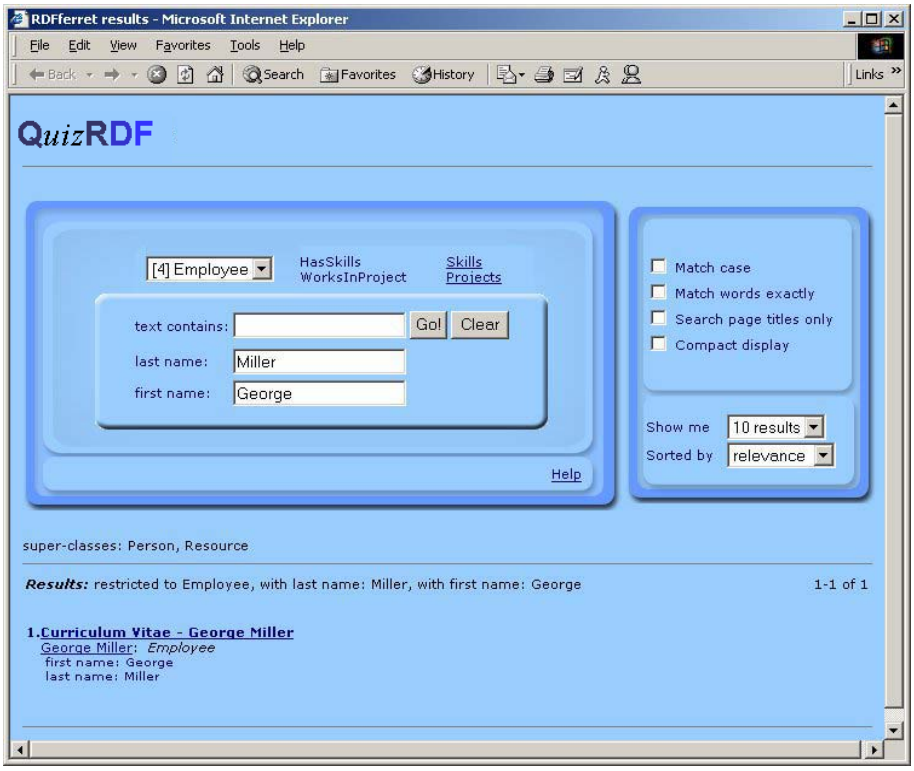

Figure 5.

4. Further Work

4.1 Technical enhancements

A number of possible enhancements to QuizRDF are in progress or under consideration and we briefly mention three of them here.

As we have seen, the resources returned from QuizRDF can be clustered based on their types (i.e. classes of which they are an instance). The proposal is then that each resulting cluster can then be 'scored' by combining some score of the individual WWW resources contained therein. (Currently, as mentioned above, QuizRDF uses a variation of the well-understood *tf.idf* scoring scheme to rank WWW resources against a user query [8]). In this way, each resource type (ontological class) can be ranked by relevance to the user query and results can be presented aggregated around resource types. Currently, we are developing a text-based way of presenting these results but there are clearly opportunities to exploit a graphical interface here also.

A second area of ongoing work stems from an obvious limitation of QuizRDF as currently implemented: namely, queries can only be made around one class. To exemplify, we can ask the queries

> "Find me all employees with last name Miller" and
>
> "Show me all instances of the class painter"

but we cannot (at least not in a single step) ask queries involving any "chaining", for example:

> "Find me all instances of class painting painted by an (instance of class) painter whose first name is Pablo"

Allowing this possibility without comprising the simple and intuitive interface of QuizRDF is the subject of ongoing research.

The third area of current work involves the implications of indexing large ontologies in QuizRDF. The issue is not one of scalability at the indexing level: the technology underlying QuizRDF has been used to comfortably index a collection of 3.5 million documents. Rather, the issue arises at the user interface level: in a heavily interconnected ontology, or one in which classes typically have a large number of properties, how can we display many classes and properties intuitively in the QuizRDF interface? Some initial work has been carried out involving use of drop-down lists rather than lists of hyperlinks but further effort is needed. There is relevant work from the HCI community that can be brought to bear on this problem, including, for example, the AlphaSlider [10] and FineSlider [11] approaches. A more radical departure in terms of the QuizRDF interface would be to adopt an approach such as described in [14], whereby search results are displayed in a two-dimensional format that uses categorical and hierarchical axes: users see the entire result set and can then click on labels to move down a level in the hierarchy.

Another topic we would like to address at some future point is the issue of creating a single interface onto two or more ontologies and their associated data. We have not yet started work in this area.

4.2 Evaluation

Several organisations (including our own) are interested in evaluating QuizRDF. We briefly mention here the use that EnerSearch intend to make of the system. EnerSearch is a virtual organisation researching new IT-based business strategies and customer services in deregulated European energy markets on behalf of its member companies. As such, EnerSearch is a knowledge creation company. In common with most WWW-based information, EnerSearch's WWW site for the use of its members currently holds weakly structured information in mixed media. EnerSearch's site is a key component for delivery of its primary role: the transfer of knowledge to its shareholders (member companies), employees and other interested parties. It is intended to carry out a study to compare the effectiveness of EnerSearch's current "text-only" search engine with the combination of free text search and structured ontological browsing which is embodied in QuizRDF.

EnerSearch have designed a detailed experiment to evaluate the advantages of QuizRDF, using both qualitative and quantitative evaluation techniques [9] and we hope to report on this in future publications.

5. Concluding remarks

Discussion with potential users as well as evidence from the information retrieval literature indicated clearly the desirability of combining RDF browsing and querying with full text search. A full text search capability means a user can enter a relatively simple initial query that essentially quickly locates them in the information space, from where further browsing and searching can proceed. Additionally, supporting full text search means that the user can access the information even at an early stage when annotations are still sparse, while the support for RDF in QuizRDF allows structured browsing of an ontology. As RDF annotations are added to the system the user will benefit from the high precision and semantic expressiveness of RDF querying. This can be seen as a ***'low threshold, high ceiling'*** approach: the user can start using QuizRDF without necessarily having to invest a lot of time in creating a rich set of annotations; while on the other hand, every newly added annotation will have an immediate effect on the system's performance and usability.

We have argued that QuizRDF's combination of RDF-based ontological browsing and searching supports a more natural and intuitive information seeking process than is available in either a search engine or a browsing tool alone.

We have described our initial implementation of QuizRDF and indicated some further directions of research and briefly discussed an ongoing evaluation of the system. QuizRDF is an early example of the much-improved information access tools that the advent of the Semantic Web makes possible.

Acknowledgements

The research described in this project was funded in part by the EU IST project OnToKnowledge (www.ontoknowledge.org), reference IST-1999-10132. Audrius Stonkus is thanked for his work on the formatting and diagrams of this paper.

References

[1] Berners-Lee, T., J. Hendler, and O. Lassila: *The Semantic Web*. Scientific American (May 2001)

[3] Davies, J., Fensel, D. & van Harmelen, F.: *Towards the Smeantic Web*, Wiley, UK (2003)

[3] Goldman-Segall, R. & Rao, S.V.: *A collaborative online digital data tool for creating living narratives*, in : Organisational Knowledge Systems, 31st Hawaii International Conference on Systems Science, Hawaii, USA (1998)

[4] Shipman, F.M., Marshall, C.C. & Moran, T.P.: *Finding and using implicit structure in human-organized spatial layouts of information*. CHI-95 (1995)
[5] Bates, Marcia J.: *An exploratory paradigm for online information retrieval* In: Brookes, B.C. (ed.) Intelligent Information Systems for the Information Society. Amsterdam, North-Holland (1986)
[6] Manber, U., Smith, M. & Gopal, B.: *WebGlimpse – Combining Searching and Browsing*. Usenix 97 Technical Conference (1997)
[7] Salton, G.: *Automatic Text Processing*. Reading, Mass., USA: Addison-Wesley (1989)
[8] Harman, D., *Ranking Algorithms*, in Frakes, W. & Baeza-Yates, R. Information Retrieval, Prentice-Hall, New Jersey, USA (1992)
[9] Iosif, V. & Ygge, F.: *On-To-Knowledge: EnerSearch Virtual Organisation Case Study*, Deliverable 28, ONToKnowledge project, www.ontoknowledge.org
[10] Ahlberg, C. & Shneiderman, B. *The Alphaslider: A Compact and Rapid Selector*, in Proc. of CHI `94, Human Factors in Computing Systems, pages 365-371, 1994.
[11] Masui, T., Kashiwagi, K. & Borden, G., *Elastic Graphical Interfaces for Precise Data*, in Proc. Chi95, 1995.
[12] Munroe, K.D. & Papakonstantinou, Y., *A Visual Interface for Browsing and Querying of XML,* in VDB5 2000, 5th IFIP2.6 Working Conference on Visual Database Systems Fukuoka - Japan May 10-12, 2000.
[13] Hearst, M., *Next Generation Web Search: Setting Our Sites*: in IEEE Data Engineering Bulletin Special issue on Next Generation Web Search, September 2000.
[14] Shneiderman, B., Feldman, D., Rose, A., and Ferre Grau, X., Visualizing Digital Library Search Results with Categorical and Hierarchial Axes, Proc. 5th ACM International Conference on Digital Libraries (San Antonio, TX, June 2-7, 2000), ACM, New York, 57-66, 2000.