Developing a Knowledge System for Information Operations

Louise Leenen, Ronell Alberts, Katarina Britz, Aurona Gerber, Thomas Meyer
Council for Scientific and Industrial Research, Pretoria, South Africa
lleenen@csir.co.za
ralberts@csir.co.za
abritz@csir.co.za
agerber@csir.co.za
tmeyer@csir.co.za

Abstract: In this paper we describe a research project to develop an optimal information retrieval system in an Information Operations domain. Information Operations is the application and management of information to gain an advantage over an opponent and to defend one's own interests. Corporations, governments, and military forces are facing increasing exposure to strategic information-based actions. Most national defence and security organisations regard Information Operations as both a defensive and offensive tool, and some commercial institutions are also starting to recognise the value of Information Operations.

An optimal information retrieval system should have the capability to extract relevant and reasonably complete information from different electronic data sources which should decrease information overload. Information should be classified in a way such that it can be searched and extracted effectively. The authors of this paper have completed an initial phase in the investigation and design of a knowledge system that can be used to extract relevant and complete knowledge for the planning and execution of Information Operations. During this initial phase of the project, we performed a needs analysis and problem analysis and our main finding is the recommendation of the use of logic-based ontologies: it has the advantage of an unambiguous semantics, facilitates intelligent search, provides an optimal trade-off between expressivity and complexity, and yields optimal recall of information. The risk of adopting this technology is its status as an emerging technology and therefore we include recommendations for the development of a prototype system.

Keywords: Information Operations, Knowledge Representation, Ontology, Query Language

1. Introduction

Businesses, governments, and military forces are increasingly reliant on the effective management of vast sources of electronic information. The type of information can be documents, images, maps, or other formats. These data sources can be used in Information Operations (IO).

McCrohan (McCrohan 1998) defines IO as "*actions taken to create an information gap in which we possess a superior understanding of a potential adversary's political, economic, military, and social/cultural strengths, vulnerabilities, and interdependencies than our adversary possesses of us*". All institutions that rely on information are facing increasing exposure to strategic information-based actions, and need to consider systems security. Most national defence and security organisations regard IO as both a defensive and an offensive tool, and some commercial institutions are starting to recognise the value of IO. In any competitive environment, an institution has to protect their strategies from competitors and gather information regarding their competitors' objectives and plans. IO include competitive intelligence, security against the efforts of competitors, the use of competitive deception, and the use of psychological operations. (McCrohan 1998).

The aim of an efficient information retrieval system is to support institutions in planning IO. Information has to be presented for processing by computers in a knowledge system such that information can be retrieved and conclusions can be drawn from existing knowledge. Information should be classified in a way that it can be searched and extracted effectively.

We present the main decisions required in the investigation and design of a knowledge system that can be used to extract relevant and complete knowledge for the planning and execution of IO and give a motivation for our main recommendation: the use of logic-based ontologies in a knowledge system for IO.

2. Intelligent Knowledge Retrieval Methods & Technologies

We describe appropriate technologies for intelligent search and retrieval of information over a range of different sources and types. The operative word here is *intelligent*, focussing on methods that will ensure maximum recall with a high level of fidelity. In other words, the aim is to get as close as is currently feasible to the ideal situation in which *all* and *only* relevant information will be returned. In order to do so, it is necessary to be more precise in deciding what it means for information to be *relevant*. The most important step in this direction is the distinction between *syntactic* and *semantic* relevance.

Syntactic relevance refers to search based on the syntactic structure of the entities to be searched, while semantic relevance is concerned with the underlying meaning of the syntactic objects being represented. Search based on syntactic relevance can be better or worse depending on some flexibility built into the search mechanisms, but this provides only for a very limited and restricted form of intelligence. To be seen as performing intelligent search in any true sense of the word, it is necessary to make use of some version of semantic relevance.

The basic assumption is that information can be accessed electronically. Information in this sense is defined very broadly: it can refer to data entries stored in database systems, or in more sophisticated structures. It can also refer to electronic documents, or an image in any of the known formats, or any one of the other numerous resources that can be stored electronically. The main reason why it is possible to allow for such a broad definition is that the methods detailed in this survey allow for a clean separation between information, the structures employed to store the information, and the methods used to access the information.

2.1 Query Languages

2.1.1 Boolean Combinations of Keywords

Keyword search is an established technology (Kalyanpur et al. 2006). The simplest form is when a list of keywords is used with the intention to locate information containing *all* keywords in the list. More flexible keyword searches can be done by using Boolean operators such as AND, OR and NOT. This kind of query language can not be used in database-style structures. A second difficulty is that searches become complex when there are large numbers of keyword hits.

2.1.2 Logic-based Query Languages

The use of logic-based languages is pervasive in database systems. It has its origins in languages such as SQL and later extensions such as the query languages for Datalog (Ceri et al. 1989) and logic programming (Lloyd 1987). These languages are all fragments of first-order logic (Ben-Ari 2008). In addition to the Boolean operators discussed in the previous section, these query languages also allow for the use of *variables*, *existential quantification* (exists), *universal quantification* (for all), and *function symbols*, and combinations of these additions in manner reminiscent of the recursive definition in the previous section. This allows us to express complex queries such as:

> "*Find all countries in Africa with a per capita income of at most $X, and with a military style government, or where there is no adherence to human rights*".

The main advantages of these types of query languages are that they allow for much more complex queries, can be used to express queries about concepts as well as individuals, and are applicable to information contained in database-style structures as well as electronic documents. However, the processing of such queries can be very complex, and is directly related to the complexity of queries. It is good practice to limit the expressivity of a chosen query language to precisely what is necessary in order to maximise the efficiency of query processing.

2.2 Information Types

It is useful to assume that information is *tagged* with the relevant components to be matched with queries. This assumption enables us to reduce the original question to a decision of how a piece of information should be tagged. A tag is a keyword associated with a piece of information. The purpose of a tag is to describe an item and to enable an electronic search to find it .

We distinction between using text or keywords as tags, and between information contained in database-style structures and electronic documents viewed as information.

2.2.1 Text as Tags
In the case of information contained in database-style structures, the only practical option is to view the information itself as its own tag. In the case of electronic documents, the simplest form of tagging is the brute force approach of using the raw text contained in a document. In a sense the document is tagged with all of its textual content. The advantage of such an approach is that it is relatively simple to implement, but this simplicity is associated with high levels of inaccuracy. In particular, this approach is bound to lead to many false positives and it does not guarantee that all relevant documents will be located. The main problem is that this is a purely syntactic approach. There is no attempt to tag documents with keywords related to the *meaning* of the document, and there is therefore no guarantee that the tags will be truly relevant to the content of the document.

2.2.2 Keywords as Tags
In contrast with using text as tags, the practice of tagging information with appropriate keywords allows for a much more flexible approach. The goal is to tag documents with keywords that are clearly relevant to the meaning of the document, ideally to tag documents with *all and only* the relevant keywords. The primary issue to be resolved here is to how to decide on the relevant keywords.

Tagging can take one of three forms: Manual tagging, semi-automated tagging, or automated tagging (Buitelaar, Cimiano 2008; Buitelaar, Magnini 2005). Current techniques are relatively good at picking out keywords related to concepts and individuals, but much work still needs to be done regarding keywords related to relationships between concepts or individuals.

Manual tagging is a good starting point however, using only manual tagging is usually not feasible, due to factors such as time constraints and the availability of domain experts. A better approach is to interleave processes for manual, semi-automated and automated tagging of documents. Automated tagging is faster but not as accurate, whereas semi-automated tagging provides better results, but is more time consuming to set up. Keep in mind that the results obtained even from manual tagging are only as good as the knowledge applied by the person(s) performing the tagging.

The good news is that tagging method lends itself to an incremental approach. One can start with a fairly course-grained tagging methodology, and refine this increasingly over time.

2.3 Information Retrieval Methods

2.3.1 Direct Retrieval
Direct retrieval is concerned with methods for extracting information stored *explicitly* in as efficient a manner as possible. This is the kind of retrieval based on indexing techniques that one would obtain from traditional database systems and from keyword searches based on syntactic relevance (Gray, Reuter 1992; Kroenke 1997). In the case of direct *document* retrieval, keywords in a query are identified and are matched directly with the keywords used to tag the document.

Direct retrieval techniques are firmly established, and are able to deal efficiently with huge amounts of information. The only drawback is the restriction on the type of information to be extracted: it has to be stored explicitly in some form.

2.3.2 Indirect Retrieval
A more sophisticated approach is to employ some kind of *indirect retrieval* where the task is to match the keywords identified in the query not just with the exact keywords with which a document is tagged, but also with *related* keywords. The hard part is to determine what constitutes *being related*. Standard approaches to indirect document retrieval are mostly still syntax-based:

- The use of synonyms using resources such as WordNet (`http://wordnet.princeton.edu/`) (Fellbaum 1998).
- Lemmatisation, the process of grouping together the different inflected forms of a word so they can be analysed as a single item (Brown 1993). For example, the verb “to walk” may appear as “walk”, “walked”, “walks”, “walking”. The base form, “walk”, is called the lemma of the word.

- Stemming, which is closely related to lemmatisation but operates on a single word without contextual information. Related words should map to the same stem, but the stem does not have to be a valid root.

A more nuanced version of indirect document retrieval involves structures able to capture and represent sophisticated relationships between entities. The more sophisticated version of indirect retrieval employs methods for performing *inference* of some kind. Indirect retrieval also includes information that can be inferred implicitly from what is stored explicitly.

The most appropriate technology able to deal with indirect information retrieval is that based on *ontologies* (Staab, Studer 2004). The following definition of an ontology is taken from Wikipedia (`http://en.wikipedia.org/wiki/Ontology_(information_science)`): "*an ontology is a formal representation of a set of concepts within a domain and the relationships between those concepts. It is used to reason about the properties of that domain, and may be used to define the domain*".

In addition to facilitating the hierarchical structuring of information from a domain of discourse, ontologies also provide the means to impose a whole variety of other constraints, which makes it a very powerful method for representing concepts, individuals, and the relationships between them. The use of logic-based ontologies is particularly apt, since it provides the means for employing powerful and efficient mechanisms for performing inference.

2.4 Ontologies and Ontology-based Engineering

In the past fifteen years, advances in technology have ensured that access to vast amounts of data is no longer a significant problem. Paradoxically, this abundance of data has lead to a problem of information overload, making it increasingly difficult to locate relevant information. The technology of choice at present is keyword search, although many argue that this is already delivering diminishing returns, as Figure 1 below by Nova Spivack (Spivack 2007) indicates. Spivack illustrates how keyword search is becoming less effective as the Web increases in size. The broken line shows that the productivity of keyword search has reached a plateau and its efficiency will decrease in future, while the dotted line plots the expected growth of the Web.

Any satisfactory solution to this problem will have to involve ways of making information machine-processable, a task which is only possible if machines have better access to the *semantics* of the information. It is here that ontologies play a crucial role. Roughly speaking, an ontology structures information in ways that are appropriate for a specific application domain, and in doing so, provides a way to attach meaning to the terms and relations used in describing the domain. A more formal, and widely used definition, is that of Grüber (Grüber 1993) who defines an ontology as a formal specification of a conceptualisation.

The importance of this technology is evidenced by the growing use of ontologies in a variety of application areas, and is in line with the view of ontologies as the emerging technology driving the Semantic Web initiative (Berners-Lee et al. 2001). The construction and maintenance of ontologies greatly depend on the availability of ontology languages equipped with a well-defined semantics and powerful reasoning tools. Fortunately there already exists a class of logics, called Description Logics (DLs), that provide for both, and are therefore ideal candidates for ontology languages (Baader et al. 2003).

The need for sophisticated ontology languages was already clear fifteen years ago, but at that time, there was a fundamental mismatch between the expressive power and the efficiency of reasoning that DL systems provided, and the expressivity and the large knowledge bases that ontologists needed. Through the basic research in DLs of the last fifteen years, this gap between the needs of ontologists and the systems that DL researchers provide has finally become narrow enough to build stable bridges. In fact, the web ontology language OWL 2.0, which was accorded the status of a World Wide Web Consortium (W3C) recommendation in 2009, and is therefore the official Semantic Web ontology language, is based on an expressive DL (`http://www.w3.org/TR/owl2-overview/`**)**.

There is growing interest in the use of ontologies and related semantic technologies in a wide variety of application domains. Arguably the most successful application area in this regard is the biomedical field (Hahn, Schulz 2007; Wolstencroft et al. 2005). Some of the biggest breakthroughs can be traced back to the pioneering work of Horrocks (Horrocks 1997) who developed algorithms specifically tailored for medical applications. Recent advances have made it possible to perform standard reasoning tasks on large-scale medical ontologies such as SNOMED CT - an ontology with more than 300 000 concepts and more than a million semantic relationships - in less than half an hour; a feat that would have provoked disbelief ten years ago (Suntisrivaraporn et al. 2007). However, a number of obstacles still remain before the use of ontologies can be regarded as having reached the status of an established technology: mainly these are issues relating to *conceptual modeling* and *data usage*.

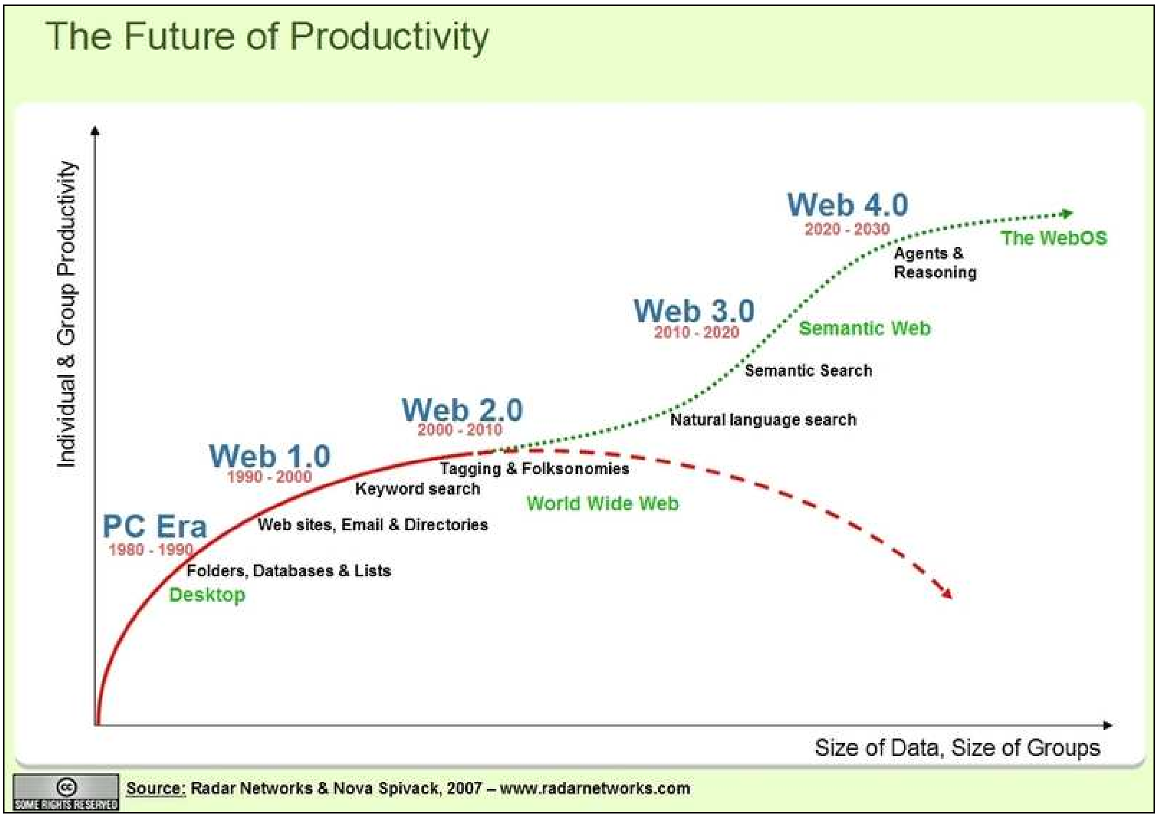

Figure 1: Productivity of Keyword Search

2.4.1 Conceptual Modeling

There are currently no firmly established conceptual modelling methodologies for ontology engineering. Although a variety of tools exist for ontology construction and maintenance (Kalyanpur et al. 2006; Sirin et al. 2007; *Protégé 2009*) they remain accessible mainly to those with specialised knowledge about the theory of ontologies. One way of dealing with this problem is to design ontology languages that are as close to natural language as possible, while still retaining the unambiguous semantics of a formal language (Schwitter et al. 2007). A related approach is to use unstructured text to automatically identify concepts and relationships in application domains, and in doing so contribute to the semi-automated construction of ontologies (Buitelaar, Cimiano 2008).

Another major obstacle is that, while most tools for ontology construction and maintenance assume a static ontology, the reality is that ontologies are dynamic entities, continually changing over time for a variety of reasons. This has long been identified as a problem, and *ontology dynamics* is currently seen as an important research topic (Baader et al. 2005; Lee et al. 2006).

2.4.2 Data Usage

Assuming that the problems relating to conceptual modeling have been solved, and that it is possible to construct and maintain high-quality ontologies, a number of stumbling blocks related to data usage still remain.

The main problem is that most available data are currently in the form of unstructured or semi-structured text, or can be found in traditional relational database systems. The rich conceptual structures provided by ontologies are therefore of little use unless ways can be found to automate, or semi-automate, the process of populating ontologies with this data. Regarding data in textual form, there have been some recent attempts to perform semi-automated instantiation of ontologies from text (Buitelaar, Cimiano 2008; Williams, Hunter 2007). With regards to the data found in database systems, it is necessary to employ *data coupling* - finding ways of linking the data residing in database systems to the ontologies placed on top of such systems (Calvanese et al. 2006). This challenge is currently being met by tools for Ontology Based Data Access (OBDA) (Rodriguez-Muro et al. 2008).

Once an ontology is populated, it becomes possible to use it as a sophisticated data repository to which complex queries can be posed, at least in principle. In practice, at least two challenges remain. The first is to perform query answering efficiently, a topic of ongoing research (Calvanese et al. 2007). The second is to go beyond purely deductive reasoning to answer queries and to be more proactive. A good example of this type of reasoning occurs during medical diagnosis, which is an instance of a form of reasoning technically known as *abduction* (Elsenbroich et al. 2007).

2.5 Tools for user Support

There is a danger that the complexity of the techniques discussed above will pose a barrier to their general uptake. Most techniques incorporate some level of familiarity with technical issues such as formal logic languages, which can be disconcerting for the more casual user. We discuss two classes of methods used to bridge the gap between users and the technology.

2.5.1 Controlled Natural Language

A controlled natural language is a suitable fragment of a natural language, usually obtained by restricting the grammar and vocabulary. This is done primarily to ensure that there is no ambiguity in the interpretation. It can also assist with a reduction in complexity. Controlled natural languages can usually be mapped to existing formal languages, typically a fragment of first-order logic.

For our purposes the translation will be to a suitable DL used to represent ontologies. Because of this mapping, controlled natural languages have a formal semantics, making them suitable as knowledge representation languages, able to support inference tasks such as query answering. The advantage of using controlled natural languages instead of their logic counterparts is that it appears to the user as if a natural language is being used. Work on controlled natural language most relevant for logic-based ontologies include Manchester OWL Syntax (Horrocks et al. 2006), Sydney OWL Syntax (SOS) (Rodriguez-Muro et al. 2008), and the Rabbit language (Hart et al. 2008).

2.5.2 Contextual Navigation

This subsection is concerned with the principles of the design and development of an intelligent query interface (Catarci et al. 2004). The interface is intended to support users in formulating queries which best capture their specific information needs. The distinctive part of this approach is the use of an ontology as the support for the intelligence contained in the query interface. The user can exploit the vocabulary in the ontology to formulate the query. Using the information contained in the ontology, the system is able to guide the user to express their intended query more precisely.

Queries can be specified through an iterative refinement process supported by the ontology through contextual navigation. In addition, users may discover new information about the domain without explicit querying, but through the subparts of a query, using classification. Work on contextual navigation is not restricted to logic-based ontology languages, but it does depend on an underlying knowledge representation language with an associated formal reasoner. In the context of ontologies, it has led to the development of a query tool as part of the European Union funded SEWASIE project (SEmantic Webs and AgentS in Integrated Economies) (`http://www.sewasie.org/`).

3. Research Methodology

We first conducted a needs analysis with our client with the aim of identifying their expectations and requirements, followed by a problem analysis where the client's domain was studied and recommendations in terms of the most appropriate technologies for their applications were made.

3.1 Needs Analysis

Needs analysis is an interactive process with the aim of extracting information from the client to understand their needs and expectations. It involves asking specific questions to the client and recording and documenting their responses. Usually several interactions are required before this process is completed.

The type of questions that were posed to our client can broadly be defined as:

- What is the reality of your domain?
- What do you do?
- What are the challenges you experience?
- What are your expectations from an information operation?

The aim of these questions is to identify the type of IO the client wants to execute, the range of required information sources and how information should be interpreted. It should also point to the type of information repositories that will be needed, and how they should be populated and updated. As a result we compiled an extensive set of derived questions. These questions depict the scope of information required by our client for an operation.

3.2 Problem Analysis
In this phase we analysed the various methodologies and technologies available for an appropriate knowledge representation system for the client's domain. A basic assumption is that all information can be accessed electronically and includes documents, images or maps, and data stored in database systems, or in more sophisticated structures.

The following three primary questions were applied to the client's domain:

- In which way will a user extract information, i.e. which query language is to be used?
- How will the type of information to be extracted be matched with the query?
- Which method will be used to retrieve the information contained in the query from the information repository?

A formal problem statement was written that includes strategic long term direction and objectives.

3.3 Findings
The main recommendation is that a logic-based ontology is to be used as the underlying technology for the retrieval system. The adoption of logic-based ontologies as underlying formalism for a knowledge representation system has a number of advantages.

The semantics of such an ontology is

- unambiguous;
- it facilitates intelligent search;
- it provides an optimal tradeoff between expressivity and complexity; and
- it can yield optimal recall of information.

The risk of adopting this technology is its status as an emerging technology. Its impressive progress in the biomedical domain lends strong support for its adoption in the IO domain, but there are presently no off-the-shelf ontologies available for IO.

The development of such an ontology that is both reliable and complete is a highly complex research endeavour. With this in mind, we recommend an incremental approach to the adoption of this technology in order to realise the long term strategic objectives outlined earlier.

The developmental recommendations for a prototype system are:

- Define a suitable sub-domain for initial development. Our client's domain is vast and complex. The recommendation is to start with a smaller, focused domain.
- The documents in the domain should be tagged. The choice of tags will depend on the ontology and the concepts used in existing information sources.
- An ontology-based search facility should be developed.

- An appropriate query language should be decided on in conjunction with a suitable user interface. which may involve controlled natural language or contextual navigation, or both.

The evaluation of a prototype system will determine the extension of the system into a comprehensive knowledge system.

4. Conclusion

In this paper we have focused on the technologies relevant for intelligent information retrieval for Information Operations. Conceptually, the survey is decomposed into three parts:

- Choices for a suitable query language;
- Type of information to be extracted;
- Methods employed for information retrieval.

Supplementary to this is a discussion on ontologies, as well as on tools for supporting users of systems for intelligent retrieval.

Our main conclusion is that the use of logic-based ontologies has the potential to be of enormous benefit in systems demanding true intelligent retrieval. However, it has to be taken into account that this is an emerging technology that will still require a substantial amount of research in order to reach maturity. The good news is that it is possible to approach matters in an incremental fashion, developing an information repository based on more traditional methods, and gradually increasing its sophistication.

References

The Protégé Ontology Editor. Available**:** http://protege.stanford.edu/. [2009, January].

Baader, F., Calvanese, D., McGuinness, D., Nardi, D. & Patel-Schneider, P. (2003) *The Description Logic Handbook: Theory,Implementation, and Applications,* Cambridge University Press.

Baader, F., Lutz, C., Milicic, M., Sattler, U. & Wolter, F. (2005) "Integrating Description Logics and Action Formalisms: First results", *AAAI 05*.

Ben-Ari, M. (2008) *MathematicalLlogic for Computer Science,* Springer.

Berners-Lee, T., Hendler, J. & Lassila, O. (2001), "The semantic web", *Scientific American,* Vol. 284, No. 5.

Brown, L. (1993) *The New Shorter Oxford English Dictionary on Historical Principals,* Vol 1*,* Oxford University Press.

Buitelaar, P. & Cimiano, P. (2008) "Ontology Learning and Population: Bridging the Gap Between Text and Knowledge", *Frontiers in Artificial Intelligence and Applications,* Vol. 167.

Buitelaar, P. & Magnini, B. (2005) "Ontology Learning From Text: Methods, Evaluation and Applications", *Frontiers in Artificial Intelligence and Applications,* Vol. 123.

Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Poggi, A. & Rosati, R. (2006) "Linking Data to Ontologies: The Description Logic DL-LiteA.", *The 2nd workshop on OWL*.

Calvanese, D., Giacomo, G.D., Lembo, D., Lenzerini, M. & Rosati, R. (2007) "Tractable Reasoning and Efficient Query Answering in Description Logics: The DL-Lite Family.", *Journal of Automated Reasoning,* Vol. 39, No. 3.

Catarci, T., Dongilli, P., Mascio, T.D., Franconi, E., Santucci, G. & Tessaris, S. (2004) "An Ontology Based Visual Tool for Query Formulation Support", *ECAI 2004*.

Ceri, S., Gottlob, G. & Tanca, L. (1989) "What you always wanted to know about Datalog (and never dared to ask.)", *IEEE Transactions on Knowledge and Data Engineering,* Vol. 1, No. 1.

Elsenbroich C., Kutz O. & Sattler, U. (2007) "A Case for Abductive Reasoning over Ontologies", *OWLED*.

Fellbaum, C. (1998) *WordNet: An Electroni Lexical Database,* MIT Press.

Gray, J. & Reuter, A. (1992), *Transaction Processing: Concepts and Techniques,* Morgan Kaufmann Publishers.

Grüber, T. (1993) "A translation approach to portable ontology specifications", *Knowledge Acquisition,* Vol. 5.

Hahn, U. & Schulz, S. (2007) "Ontological foundations for biomedical sciences", *Artificial Intelligence in Medicine,* Vol. 39, No. 3.

Hart, G., Dolbear, C. & Johnson, M. (2008) "Rabbit: Developing a Control Natural Language for Authoring Ontologies", *5th European Semantic Web Conference*.

Horrocks, I. (1997) *Optimising Tableaux Decision Procedures for Description Logics*, University of Manchester.
Horrocks, M., Drummend, N., Goodwin, J., Rector, A., Stevens, R. & Wand, H. (2006) "The Manchester OWL Syntax", *OWL Experiences and Directions Workshop*.
Kalyanpur, A., Parsia, B., Sirin, E., Cuenca-Grau, B. & Hendle, J. (2006) "Swoop: A Web Ontology Editing Browser", *Journal of Web Semantics*, Vol. 4, No. 2).
Kroenke, D.M. (1997) *Database Processing: Fundamentals, Design, and Implementation,* Prentice-Hall.
Lee, K., Meyer, T., Pan, J.Z. & Booth, R. (2006) "Finding Maximally Satisfiable Terminologies for the Description Logic ALC", *Proceedings of AAAI 06*.
Lloyd, J.W. (1987) *Foundations of logic programming,* Springer-Verlag, New York.
McCrohan, K.F. (1998) "Competitive Intelligence: Preparing for the Information War", *Long Range Planning,* Vol. 31, No. 4.
Rodriguez-Muro, M., Lubyte, L. & Calvanese, D. (2008) "Realizing Ontology Based Data Access: A plug-in for protoge", *ICDE Workshops*.
Schwitter, R., Cregan, A. & Meyer, T. (2007) "Sydney OWL Syntax - towards a Controlled Natural Language Syntax for OWL 1.1.", *OWL Experiences and Directions, Third International Workshop*.
Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A. & Katz, Y. (2007) "Pelet: A practical OWL-DL reasoner", *Journal of Web Semantics,* Vol. 5, No. 2.
Spivack, N. (2007). Available: http://novaspivack.typepad.com/nova_spivacks_weblog/2007/03/beyond_keyword_.html. [2010, November 2010]
Staab, S. & Studer, R. (eds) (2004) *Handbook on Ontologies*, Springer.
Suntisrivaraporn, B., Baader, F., Schulz, S. & Spackman, K. (2007) "Replacing SEP-Triplets in SNOMED CT using Tractable Description Logic Operators", *AIME*.
Williams, M. & Hunter, A. (2007) "Harnessing ontologies for argument-based decision-making in breast cancer", *International Conference for Tools with Artificial Intelligence*.
Wolstencroft, K., Brass, A., Horrocks, I., Lord, P., Sattler, U., Stevens, R. & Turi, D. (2005) "A little semantic web goes a long way in biology", *International Semantic Web Conference*.