European Symposium on Computer Aided Process Engineering – 15
L. Puigjaner and A. Espuña (Editors)

Multi-Agent Systems for Ontology-Based Information Retrieval

R. Bañares-Alcántara[a*], L. Jiménez[b] and A. Aldea[c]

[a]Department of Engineering Science, Oxford University
Parks Roads, Oxford OX1 3PJ, UK

[b]Department of Chemical Engineering and Metallurgy, University of Barcelona
Martí i Franquès 1, Barcelona 08028, Spain

[c]Department of Computing, Oxford Brookes University
Wheatley Campus, Wheatley, Oxford OX33, UK

Abstract

The Web offers a huge amount of valuable information, but it is very hard and time consuming to retrieve thousands of web pages related to a concept, filter the relevant ones, analyse this information and integrate it in a knowledge repository. This paper describes one component of a knowledge management platform (*h-TechSight* project) that performs these tasks, the multi-agent search module (MASH). MASH employs a domain ontology to search for web pages that contain relevant information to each concept in the domain of interest. The search is then constrained to a specific domain to avoid as much as possible the analysis of irrelevant information.

Keywords: ontology; multi-agent system; knowledge retrieval.

1. Introduction

A good use of knowledge management practices can greatly benefit knowledge intensive industries, such as chemical process industries. Maintaining an up-to-date knowledge of the domain is of capital importance for those industries. The WWW offers a huge amount of information, but it is impossible for a person to retrieve thousands of web pages related to a concept, filter the relevant ones, analyse their content and integrate it in the company knowledge repositories [Batres et al., 2002]. Knowledge management tools can help by providing tools that automatically update technological domains, and monitor and assess how products, services, and technologies evolve, emerge, mature or decline.

Furthermore, engineers typically identify the evolution of their disciplines by reading journals, attending conferences or by hearsay. All this information can be found nowadays on the web, but it is weakly structured, scattered, distributed and impossible to analyse manually. Traditional search engines allow users to retrieve information by combining keywords. This type of search can cause several problems: the number of

* Author to whom correspondence should be addressed: rene.banares@eng.ox.ac.uk

pages retrieved may not be manageable; some of the retrieved documents are irrelevant while some of the relevant documents may have not been retrieved.
Fensel (2001) argued that the performance of a search engine could be improved by using ontologies. In its conventional form, an ontology can be seen as a representation of the concepts which are relevant to a particular domain. Ontologies provide a semantic view that helps to sort out web pages with relevant information about a concept from web pages that contain data with just syntactic similarities to the concept.
The aim of the EU research project *h-TechSight* (h-TechSight, 2001) is the construction of a knowledge management platform (KMP) (Stollberg et al., 2004; Kokossis et al, 2005), which can be used by knowledge-intensive industries to keep a dynamically updated knowledge map of their domain. This paper describes in detail one of the main components of the KMP, the MASH search module, which main task is to find web pages with relevant information about a predefined field represented by a *domain ontology* (Aldea et al., 2003).

2. The Multi-Agent Search Engine (MASH)

The search engine requires a domain ontology to perform the search, so the user must generate that ontology or use an existing one to start the procedure. The implementation of the search module is based on the agent technology (Wooldridge, 2002) where several software agents work together in an asynchronous, concurrent and intelligent way that can be distributed among several computers.

2.1. *Domain ontology*

The search is driven by the domain ontology which is represented by a hierarchical taxonomy of concepts. Every concept (class) is connected to a parent concept (super-class) and thus a class and all its ancestors define a class path (for example, in Figure 1: Biosensor\Application\Environment\Air analysis). Every class (e.g.. air analysis) contains a set of slots which represent the properties and characteristics that are important for this specific class in the general domain of interest (e.g. biosensors). Every class also inherits all slots defined in its ancestors (Fensel, 2001). Figure 1 depicts a part of one *domain* ontology (biosensor ontology) used in this work. For instance, the concept biosensor application in health care is represented by the subclass “health-care” within this class, the user decided to include two slots “researcher-field” and “research-topic”.
During the search process, the difference between classes and properties is that classes define the search domain, while properties are used to evaluate to what extend the retrieved pages have the sort of information required by the user.
The use of synonyms in the definition of classes and slots extends the domain ontology with the possibilities of having alternative terms as, for example, "domain" and "field", acronyms as "computer fluid dynamics" and "CFD", chemical formulas as “sodium sulphate” and “Na_2SO_4” or language differences as "generalisation" and "generalization".

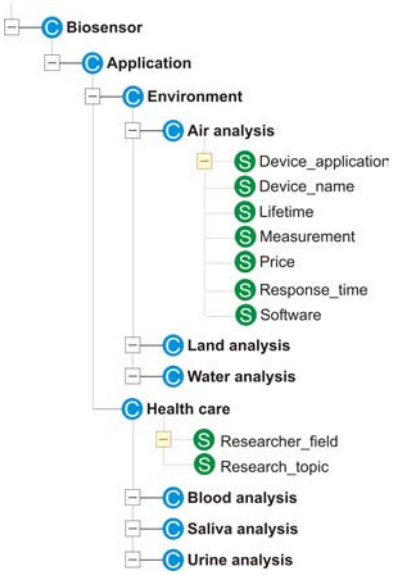

Figure 1. Biosensors ontology as a search domain.

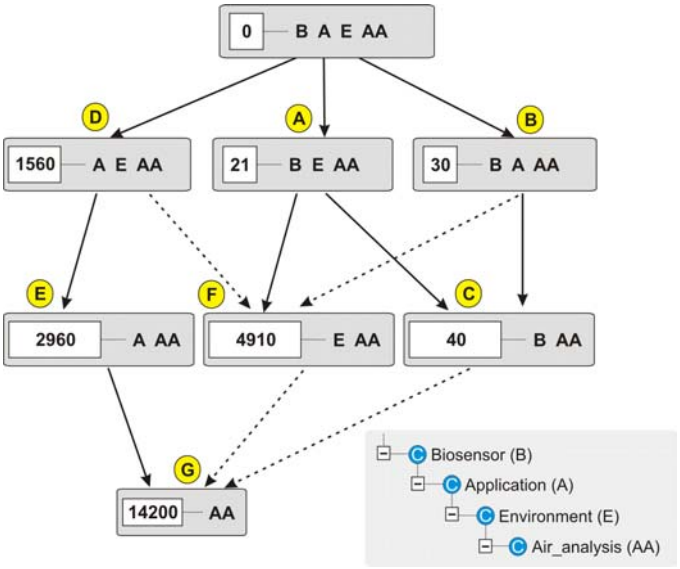

Figure 2. Best first search developed by the weight agent.

2.2 The search process

Once the *domain ontology* and the search parameters have been defined by the user, the multi-agent search process starts. This task combines several stages: splitting the *domain ontology*, retrieving the web pages, rating and filtering the retrieved pages, and the classification of the results.

The domain ontology is first dividing into query ontologies. A *query* ontology consists of a class with all its slots (own and inherited) and the class path. The query ontologies

are used by the search agents and as a result, for every search a set of web pages is retrieved. The constraints *web site* (e. g. yourcompany.net or mycompany.com), *language* (e. g. German) and *country* (e. g. mx or jp), are also considered. Moreover, MASH is able to search databases or documents available on the intranet (*.pdf or *.doc). If the number of web pages retrieved does not reach the *MaxLinks* parameter predefined, the system raises a complementary process, based on an expansion tree of the initial query, as can be depicted in Figure 2 where query for the class air_analysis has been expanded. The query building process is as it follows: each node of the tree is expanded with sub-nodes representing queries where one of the keywords has been removed, except the name of the current class (right bottom side of Figure 2). When one of the parents of "air analysis" (i. e. "environment", "application", or "biosensor") is removed from the initial query, the nodes A, B, E are respectively expanded, while "air analysis" (AA) is in all three sub-nodes. The number of web pages found for each node is depicted in the white box inside each node in Figure 2.

An alternative way to extend the number of web pages to reach the *MaxLinks* parameter is to use the *Depth* parameter. This value indicates to what extend the search process takes into consideration the links contained in the retrieved pages. So, if *Depth* is 0 only pages recovered by the search engine are considered. But if *Depth* is set to 1, also pages directly linked from *Depth* 0 pages are retrieved, rated and filtered. *Depth* values above two are not recommended because the time increases exponentially.

The final stage of the search process is rating and filtering the retrieved pages according to their relevance to the *query ontology*. The rate is calculated with the function:

$$R_c(p, A) = \frac{\text{number of attributes encountered}_{(p,A)} \cdot 100}{\text{total number of attributes}_{(A)}} \quad (1)$$

where p is the web page recovered for a class *C* and *A* is the set of attributes (inherited or not) of *C*. $R_C(p,A)$ defines the relevance of the web page p with respect to the class *C*. After normalisation (range [0, 1]), and during the filtering step, this value is used to discard pages below the *Threshold* parameter, and finally, to rank pages.

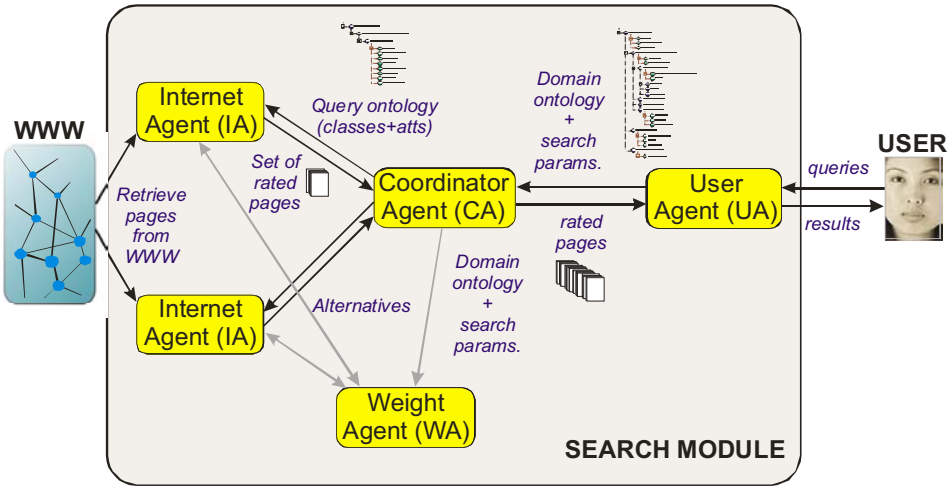

Figure 3. Multi-agent system architecture.

In the final step, all pages obtained for all classes of the *domain ontology* are combined into a single information ontology.

2.3. MASH Architecture

The search process was implemented in the Multi-Agent JADE environment (Bellifemini, 2001). Four different types of agents were identified (Figure 3): one *user agent* (UA), which interacts with the user, n *internet agents* (IA) which access, retrieves, rates and filters the information from the web, one w*eight agent* (WA) that supports the search and supplies alternative queries when requested, and one *coordinator agent* (CA) that rules the overall process, particularly during splitting and amalgamation of the tasks performed by the IA. The way these agents collaborate can be explained following the interactions contained in Figure 3, a more detailed description of the platform can be found in Moreno et al. (2004).

- The user interacts with the system through the UA, providing the *domain ontology* and setting up the parameters described in section 2.2.
- The CA receives and divides the *domain ontology* into the *query ontologies* that are spread out across all the available IAs (configurable with *Internet Agents*).
- Each IA uses the search engine (configurable with the *SearchEngine*) and the semantic information of the *query ontology* received from the CA to filter and to sort the web pages, and sends them back to the CA. If the IA does not recover enough pages, the WA supplies alternative queries to complete the list.
- In this case, the WA explores the expansion tree and proposes the most restrictive query. For instance the alternative query at level 2 corresponds to node A (Figure 2).
- Finally, the CA waits for all the IAs to supply the results or until the *Deadline* is reached. Then the CA amalgamates all the results and sends a unified ontology to the UA, who shows the results to the user.

Both the input and the output of the search process are handled using a web interface where the user can edit, create, retrieve, import and export ontologies as Resource Description Format files (W3C, 2001).

3. Tests and Results

The MASH search engine has been extensively used in different domains such as biosensors, chemical engineering, and process engineering employment agencies (Aldea et al, 2003). A more detailed analysis of the search module to study the evolution of the relevance of the retrieved pages depending on the input parameters has been produced in the field of biosensors. Some of these studies show that the number of pages about biosensors that have been updated during the last three months are above 94800. For such amount of information and dynamics, the use of automatic intelligent search systems, such as the one described in this paper, is compelling. One first benefit is to reduce the number of retrieved pages to a manageable quantity, by keeping for each concept of the ontology only those pages which are the most relevant to the concept and its attributes, within the context of the ontology.

Three main outcomes arise from the analysis of the results. Firstly, the lack of attributes in a class produces a deceptive appearance of high relevance in comparison to other classes with attributes. A deeper analysis shows that the reason is that the search is less

demanding as the number of attributes decreases. Secondly, the average quality of the pages has a slight descending trend as the number of pages required increases. This result confirms that when MASH is forced to recover more pages, and thus new pages, which are progressively less relevant, are considered. Finally, it is also interesting to observe that the average relevance of the pages associated with a class is typically higher than the average relevance of the pages of their subclasses, but this difference is smaller as we move deeper in the ontology hierarchy. The reason is twofold, when the classes become more specific (deep in the hierarchy) they are more restrictive and as consequence less good pages are found. However, if the number of recovered pages is kept constant for all the classes, those classes that are more restrictive are forced to include less relevant pages, and the average rate descends.

4. Conclusions

MASH, a multi-agent search engine has been described in this paper. MASH is able to detect web pages related to a *domain ontology* and calculate the relevance of each one. Additionally, the search parameters allow the user to control some aspects of the search (*Deadline*, *SearchEngine*, *MaxLinks*, *Threshold* and *Depth)*. The system was tested to analyse the relevance of the pages as the user parameters are modified. The multi-agent system has been developed to make the whole process asynchronous, concurrent, intelligent and distributed.

References

Aldea, A., R. Bañares-Alcántara, J. Bocio, J. Gramajo, D. Isern, L. Jiménez, A. Kokossis, A. Moreno and D. Riaño, 2003, An Ontology-based Knowledge Management Platform, Workshop IIWEB'03 at IJCAI'03, Acapulco, México: 177-182.

Batres, R., R. Chatterjee, R. Garcia-Flores, C. Krobb, A. Yang and X. Z. Wang, 2002, Software Agents, in B. Braunschweig and R. Gani, Software Architectures and Tools for Computer Aided Process Engineering, Amsterdam, Elsevier 11: 455-484.

Bellifemine, F., A. Poggi and G. Rimassa, 2001, Developing Multi-Agent Systems with a FIPA Compliant Framework, Software Practice and Experience, 31: 103-128.

Fensel, D., 2001, Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce, Heidelberg, Germany.

h-TechSight, 2001, IST Project (IST-2001-33174). prise-serv.cpe.surrey.ac.uk/techsight (Access: January 2005).

Kokossis, A., R. Bañares-Alcántara, L. Jiménez and P. Linke, 2005, *h-Techsight*: a Knowledge Management Platform for Technology Intensive Industries, European Symposium on Computer Aided Process Engineering (ESCAPE-15), Barcelona, Spain.

Moreno, A., D. Riaño, D. Isern, J. Bocio, J. Sánchez and L. Jiménez, 2004, Knowledge Exploitation from the Web, 5th International Conference on Practical Aspects of Knowledge Management (PAKM'04), Viena, Austria.

Stollberg, M., A. Zhdanova and D. Fensel, 2001, *H-TechSight*: a Next Generation Knowledge Management Platform, J. of Information and Knowledge Management, 3 (1): 47-66, 2004.

W3C, 2001, Resource Description Framework.

Wooldridge, M., 2002, An Introduction to Multiagent Systems, John Wiley and Sons Ltd, New Jersey, USA.

Acknowledgement

This work has been funded by the *h-TechSight* EU project (IST-2001-33174).