elpub2002 Proceedings — J. A. Carvalho et. al. (Eds.) — VWF Berlin, 2002

Advanced Meta-search of News in the Web

Rubén Tous and Jaime Delgado

Universitat Pompeu Fabra (UPF), Departament de Tecnologia,
Pg. Circumval·lació, 8. E-08003 Barcelona, Spain
{ruben.tous, jaime.delgado}@tecn.upf.es

Abstract. In specific domains, such as newspaper news, virtual libraries, videos or music repositories, the available specialised search engines use to offer to the users more complex interfaces than the generic ones. These interfaces allow to specify constraints about specific features of the resources being searched, as the date of an article, the price of a book or the duration of a movie. On the other hand we have traditional specialised meta-search engines, those engines that help users in their search process by automatically querying a set of specialised search engines available for a given domain. This kind of engines usually offer only a "one-field" interface to the user because of the difficulty to feature the interface particularities of each underlying specialised engine. Our approach allows to overcome this limitation without doing any presumption over the existing technologies. We have developed a practical application, an advanced news meta-search engine that could be easily adapted to cover other domains.

1 Traditional Search Limitations

When we are looking for some information or content, in any digital environment, we have two possible alternatives: We can explore one by one all the existing objects inside the set of interest – that in the case of the Web could take probably more than a million years – or we can use a search application that allows us to express constraints about the properties of the objects that we are seeking, using some kind of language as for example SQL in the context of databases. The more expressiveness the language has the more precision the query will have.

Traditional search over the Internet is usually performed using applications known as 'search engines'. These systems seek a list of keywords among the textual content of the Web (HTML, PDF, etc.). The concordance of a resource with the query depends on the times keywords are present and also on their relative and absolute position. Traditional search engines do not explore the Web directly, they use databases previously loaded with indexes. Programs called 'spiders' retrieve and parse the textual content of the web pages in a process known as 'web crawling', responsible of maintaining the content of the indexes.

The traditional search engines user interface (Fig. 1) consists of a single text field where users can enter a sequence of keywords and boolean operators to constraint how these keywords must be searched. Because common users are not programmers most of the search sites offer an "advanced search" page to facilitate an alternative way in boolean queries.

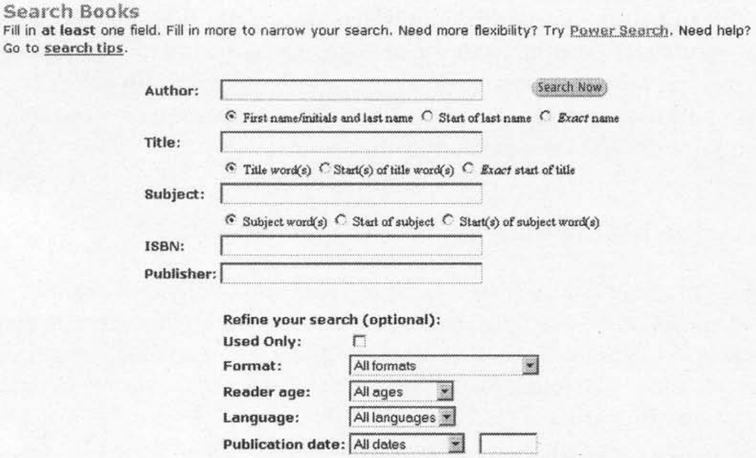
Fig. 1. *Traditional search engine user interface: Google.*

Once the search is finished the search engine shows to the user a results page, where it lists the web resources where the keywords have been found. The list is showed in descendent order, from the best result to the worst according to the criteria described before. The items of the results list, in this kind of search, contain few information about the resource described: information about the title, a short description, the size and maybe the author.

1.1 Specialised Search

In specific domains, as newspaper news, virtual libraries, videos or music repositories, the available search engines use to offer to the users more complex interfaces than the generic ones. These interfaces allow to specify constraints about specific features of the resources being searched, as the date of an article, the price of a book, etc. The results page of a specialised search engine it is quite similar to the results page described in the previous section, but it provides more information about each item in the list (Fig. 2).

Engines of different domains, as videos, music or games for example, will use a different set of attributes to describe each matching result, but even engines of the same domain, books in this case, will probably use a similar but not equal set of attributes. This is the main drawback that constraints the functionality of the



Search Books

Fill in **at least** one field. Fill in more to narrow your search. Need more flexibility? Try Power Search. Need help? Go to search tips.

Author: [Search Now]

● First name/initials and last name ○ Start of last name ○ *Exact* name

Title:

● Title word(s) ○ Start(s) of title word(s) ○ *Exact* start of title

Subject:

● Subject word(s) ○ Start of subject ○ Start(s) of subject word(s)

ISBN:

Publisher:

Refine your search (optional):

Used Only: □

Format: All formats

Reader age: All ages

Language: All languages

Publication date: All dates

Fig. 2. *Search interface of Amazon.*

existing specialised meta-search engines, as we will discuss later, and one of the targets of our research work.

1.2 Meta-search

The increasing number of search engines has motivated the apparition of new systems that help users finding information in the Internet by automatically querying a set of available search engines. These systems are called meta-search engines. From the users point of view traditional meta-search engines have the same interface and functionality as normal ones, but it is commonly accepted that they are slower.

Meta-search engines have to face the problem of querying applications designed for human interaction with interfaces as those described above. Some initiatives have appeared to define a standardized and machine-friendly access point to Web search systems, but the success of these approaches is constrained by the fact that search service providers are reluctant of other systems taking unrestrained profit of their work. However, the inexistence of machine-friendly interfaces cannot avoid the exploitation by third parties of the information harvesting effort of the existing search engines, mainly because they use browsers as a presentation layer, with exposed HTTP requests and HTML results pages. This leaves a door open to other applications to act as browsers and launch queries against them. So the task of meta-search can be divided in two main sub-problems:

1. How to query each search engine.
2. How to obtain the information from each results page.

The necessity to feature each engine interface, overall considering the lack of collaboration, is very time-consuming and cumbersome, and no one can guarantee that the interfaces will remain unchanged. This makes existing meta-search engines very difficult to maintain, and the uncertainness about their update state reduces their public acceptance.

1.3 Specialised Meta-search

If in the field of generic search we can find the figure of the meta-search engine, in the field of specialised search happens exactly the same. There exist some meta-search engines designed to launch queries against a set of specialised search engines of the same domain. Currently there exist specialised meta-search systems in practically every possible area.

As said before, specialised search engines provide complex interfaces to perform accurate queries constraining the particular features of the target resources. Surprisingly, traditional specialised meta-search usually offer only a "one-field" interface to the user (Fig. 3). The origin of this limitation lies in the difficulty to feature the interface particularities of every underlying specialised engine. To provide a richest interface it would be necessary to map the interface semantics with the semantics of every target engine, a hard task especially if we consider that the interfaces could change.

To overcome this limitation is one of the targets of our research work concerning this area, as it will be further explained.

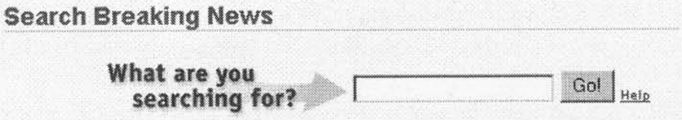


Fig. 3. *Specialised meta-search user interface: Search.com.*

2 Metadata

Metadata is machine understandable information about multimedia resources or other things, as the title and author of a photo, the date of an article or the price of a book [1].

Metadata can be applied to resource discovery, cataloguing, content rating, intellectual property rights, or knowledge sharing between software agents. Metadata has existed ever, but lately there have been some initiatives to generalize its use in

Web resources by defining standard attribute sets (IEEE LOM [2], Dublin Core [3], etc.) and standard notations as RDF [4] (the Resource Description Framework), a language based on XML [5] for describing metadata.

This new environment has originated the apparition of new search engines in the research field that still have no practical relevance because they do not offer solutions for today's Web, but that must be mentioned here [6, 7].

It is important to clarify here that in this document the use of 'metadata' refers to any information about a resource – data about data –, no matter the form it takes or the kind of resource it is (it needn't to be a web page).

3 Our Approach

Our research group has been working lately in practical solutions to improve the capabilities of meta-search engines [8]. It is not a quantitative approach, since we do not pretend to reduce the search time or to amplify the search field, but a qualitative approach. The target is to provide users – human or agents – with a search interface that allows to express unrestricted search criteria over any existing resource in the Internet, without doing any presumption over the existing technologies – as other approaches do, as the Open Archives Initiative [9]–. To achieve these goals we have defined a new strategy to design meta-search engines. This strategy is based on five main ideas:

1. Common metadata specification: The specification of a selected common set of properties – metadata – of the objects targeted by the search. This specification could be formalised using XML DTDs, XML Schemas or RDF Schemas for example.
2. User-interface independent query language: The specification of a generic query language that will be the entry point to the meta-search engine. It is not necessary to reinvent the wheel, if we assume that results will come in some XML form, W3C's XQuery [10] language will suffice – or RQL if we are using RDF–. The language needn't to be known by human users because it could be distilled from human-friendly interfaces.
3. Human/machine maintainable XML descriptors: The use of XML descriptors to feature the 'hostile' underlying engines interfaces, to facilitate it is generation and maintainability by human administrators or learning agents.
4. Mapping: The XML descriptors should allow to map the generic queries of the users (formalized in the language mentioned above) to the specific interfaces of the underlying engines. These descriptors should also be used to map the heterogeneous results obtained to the generic set of metadata. The homogeneous results obtained could be formalized using XML or RDF. Some questions arise here, as what happens with search conditions that cannot be mapped to some engines, or what must be done with results where not all the properties were

defined – specially the properties referenced in some of the search conditions –. The following point will answer these questions.

5. Reprocessing: The key aspect of our strategy is the reprocessing of the results. Because some of the conditions expressed by the generic user query probably cannot be mapped to all the underlying engines, it is necessary to reprocess the query over the obtained results, once they have been normalised. Because the user query arrives to the system in the form of a standard query language (XQuery, RQL, etc.) this stage can be performed by simply executing the respective query processor over the obtained results. This step guarantees that the results returned to the user are coherent with the conditions expressed in the initial query.

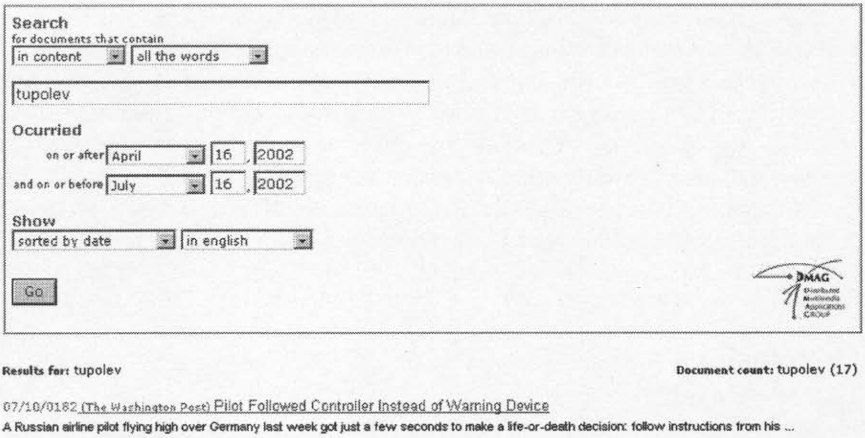
Our approach can be applied to any kind of search over the Web, but it becomes specially appropriate when it is applied to specialised meta-search. The reason is that, in despite of that the specialised search engines of the same domain use to share similar and rich sets of metadata, the traditional specialised meta-search has not found till now a way to exploit it, unless by establishing partnerships and specific protocols with the underlying engines administrators.

3.1 A Practical Application: Advanced News Meta-search Engine

We have applied our ideas in the development of an advanced meta-search engine specialised in newspaper news [11]. In this domain there exist thousands of commercial and non-commercial traditional search engines, and also hundreds of available meta-search applications. The most part of the newspapers with presence in the Web offer search services in their sites. All these engines are the potential information sources of our system, but each one of them uses a different set of parameters in the queries and a different results page format.

Our objective is to offer to the user a generic interface that allows to specify unrestricted conditions over the set of common properties that we have selected in this domain (headline, author, date, section, page, newspaper and language). To achieve this target, once selected and formalized – we are currently using a XML DTD – the subset of metadata of interest, the next step is to analyse the interface of every engine to obtain information about the query method. We use XML descriptors to describe how to map each specific set of query parameters to the generic common properties selected. We plan to use learning agents to perform this operation periodically because the interfaces of the engines could change over time. As a part of the interface featuring we must also acquire information about the results page, that will be used during the parsing process.

Once we have a mechanism to feature the engines interfaces, we can design the interface of the meta-search engine. We have selected XML messages (SOAP [12]) containing XQuery sentences and HTTP protocol. This interface is open and can be used by third-parties to develop independent clients -user interfaces or agents- However, to demonstrate the functionality of the system, we have developed our own interface (Fig. 4).



Search
for documents that contain
in content | all the words

tupolev

Ocurried
on or after April 16 , 2002
and on or before July 16 , 2002

Show
sorted by date | in english

Results for: tupolev

Document count: tupolev (17)

07/10/0182 (The Washington Post) Pilot Followed Controller Instead of Warning Device
A Russian airline pilot flying high over Germany last week got just a few seconds to make a life-or-death decision: follow instructions from his ...

Fig. 4. *DMAG's News Advanced Meta-search Engine user interface.*

The criteria specified by the user is translated to XQuery and sent to the meta-search engine. The engine maps the parts of the query – at least those that are possible – to each underlying engine interface and then launches all the searches in parallel. The results obtained are heterogeneous and must be parsed and mapped to the common set of properties.

Because no one can guarantee that all the criteria have been mapped to all the engines, the results – now homogeneous and serialized in XML – must be reprocessed. This reprocessing is easily performed in the server only by using a XQuery processor with the XQuery received as the input.

4 Agents and The Semantic Web

Until now the most part of the existing services in the Web have been designed to be accessed by human users. In the future it is supposed that software agents – maybe mobile, maybe intelligent – will access Web services and contents autonomously. This new context is commonly known as the 'Semantic Web'. A lot of research work is being done to establish the necessary conditions required by this new environment [13–15], some of them focusing on the specification of standards for machine-understandable content, some of them in the definition of protocols and languages to enable interaction between agents and services. We believe that the information search and retrieval will be one of the key services in the new era of the Web, so we think that machine-friendly interfaces will be needed to allow agents to communicate with the new search engines.

Our approach fits in this new context, because it suggests the use of a standard machine-understandable query language, like XQuery or RQL, to establish an interface-independent middle-layer between clients and the service, and the use of XML or RDF when referencing the metadata of the resources being searched. Other research projects in this field offer similar features, but the added value of our proposal lies in the fact that we are defining a strategy to adapt a human oriented service, as a search engine, to a machine-friendly service without forcing changes in the underlying technology. It is like a kind of adaptor that enables that an existing traditional service can be accessed by software agents in a transparent way.

5 Conclusions

Nowadays the Web has become the first place where people goes when they need to find some information. Surprisingly, and in concordance with what we have exposed in this document, we can affirm that the functionality of the current search systems of the Web is very limited, overall in comparison with other digital environments.

Today, the 'keywords paradigm' consisting in that one types some words in a text field and press the 'search' button, satisfies the necessities of the most part of the people, and probably the average Web user does not want to hear nothing about new search interfaces. However, the Web is growing exponentially, and also the need for information, and soon the results obtained from a query based on a list of keywords will be unmanageable, and new and faster search mechanisms will be needed. Furthermore, in the short term, the most part of the non-textual resources of the Web (images, videos, music, etc.) will be enriched with some kind of metadata, supporting new standards as MPEG-7 [16]. The queries targeting these resources must be capable to express complex conditions about properties and attributes. In the long term, the 'Semantic Web' will require strategies to adapt the existing human-oriented search services to enable its use by software agents without traumatic impact in the underlying technologies.

Our approach targets all these challenges without making assumptions of the success of some standard or protocol.

References

1. Tim Berners-Lee. Metadata Architecture.
   http://www.w3.org/DesignIssues/Metadata
2. IEEE LOM http://ltsc.ieee.org/wg12/
3. Dublin Core Metadata Initiative http://dublincore.org/
4. Resource Description Framework (RDF) http://www.w3.org/RDF/
5. XML http://www.w3.org/XML/
6. Describing and retrieving photos using RDF and HTTP
   http://www.w3.org/TR/photo-rdf/
7. Web Search Environments (WSE) http://wse.search.ac.uk/demo.html
8. Enric Peig, Jaime Delgado, Ismael Pérez. Metadata interoperability and meta-search on the web. Proceedings of the International Conference on Dublin Core and Metadata Applications 2001. (DC-2001)
   http://www.nii.ac.jp/dc2001/proceedings/product/paper-37.pdf
9. XQuery http://www.w3.org/TR/xquery/
10. Hussein Suleman. Enforcing Interoperability with the Open Archives Initiative Repository Explorer. Proceedings of the ACM/IEEE Joint Conference on Digital Libraries, Roanoke VA, 2001. http://www.dlib.vt.edu/projects/OAI/reports/jcdl_2001_paper_repository_explorer.pdf
11. News Searcher Application http://hayek.upf.es:9080/msApp/news.jsp
12. SOAP http://www.w3.org/TR/SOAP/
13. RDF Metadata and Agent Architectures
    http://www.objs.com/workshops/ws9801/papers/paper056.html
14. Retsina Semantic Web Calendar Agent http://www.daml.ri.cmu.edu/Cal/
15. Eric J. Glover et al. Architecture of a Meta-search Engine that Supports User Information Needs. Proceedings of the Eight International Conference of Information Knowledge Management (CIKM-99)
16. Moving Picture Experts Group. (MPEG-7) Multimedia Content Description Interface. http://mpeg.telecomitalialab.com/