A literature-based approach to annotation and browsing of Web resources

Miguel A. Sicilia,* Elena García,* Ignacio Aedo,* Paloma Díaz*
*Department of Computer Science, Carlos III University, Madrid, Spain
*Department of Computer Science, University of Alcalá Madrid, Spain

Abstract

The emerging Semantic Web technologies critically depend on the availability of shared knowledge representations called ontologies, which are intended to encode consensual knowledge about specific domains. Currently, the proposed processes for building and maintaining those ontologies entail the joint effort of groups of representative domain experts, which can be expensive in terms of co-ordination and in terms of time to reach consensus. In this paper, literature-based ontologies, which can be initially developed by a single expert and maintained continuously, are proposed as preliminary alternatives to group-generated domain ontologies, or as early versions of them. These ontologies encode domain knowledge in the form of terms and relations along with the (formal or informal) bibliographical resources that define or deal with them, which makes them specially useful for domains in which a common terminology or jargon is not soundly established. A general-purpose metamodelling framework for literature-based ontologies - which has been used in two concrete domains - is described, along with a proposed methodology and a specific resource annotation approach. In addition, the implementation of a RDF-based Web resource browser - that uses the ontologies to guide the user in the exploration of a corpus of digital resources- is presented as a proof of concept.

Introduction

The exponential growth of the Web has drastically changed the availability of electronic information, but, somewhat paradoxically, this success has also made it increasingly difficult to find and organize information. It is a well-known fact that users often become overwhelmed by the number of items retrieved by a simple query in a search engine, and that the precision of those results is in many cases fairly inadequate, resulting in low effectiveness, as reported, for example, in (Gordon and Pathak, 1999). The so-called Semantic Web is a relatively new research direction (Ding et al, 2002) aimed at overcoming this and other problems by providing machine-readable semantic descriptions to Web resources. These descriptions are based on ontologies, a form of knowledge representation developed within the Artificial Intelligence community. An ontology can be defined (according to Gruber (1993)) as a formal (i.e., machine-understandable), explicit specification of a shared (i.e., consensual, accepted by a group) conceptualization. Each ontology is focused on a specific domain, which can be of a very diverse nature. For illustration purposes, in the DAML Ontology Library we can find - among a large variety of them - ontologies about "beers and brewery" (including classes like lager or malt), "bioinformatics" (including
The process of attaching these semantic descriptions to existing or newly created Web resources is called **annotation**, and it essentially involves linking in some way a Web page (or an element inside it) to a number of terms or classes in one or several ontologies, which are defined in a Web-enabled ontology definition language such as DAML+OIL (Fensel, 2002). Early annotation approaches involved the inclusion of mark-up elements inside HTML pages (embedded or internal annotation) - see, for example (Benjamins & Fensel, 1998). However, a more convenient approach we call **external** annotation can be used instead. External annotation uses a separate physical storage for annotated resources, enabling the annotation of those which their source is not accessible, and not placing the burden of downloading additional mark-up in applications that are not ready to process it. Once the annotations are available, search engines, browsers, recommenders and similar Web services or applications can be built to take advantage of the semantic descriptions (Lu et al, 2002), in combination with other well-known hypermedia technologies (Ossenbruggen et al, 2001).

In any case, annotation must always be based on **consensual** knowledge in an open system like the Semantic Web is supposed to be. The just described applications assume that shared and commonly agreed ontologies (made by "ontogroups") are available for the information domains of the resources being annotated, thus making the role of Ontology Library Systems (Ding & Fensel, 2001) a critical success factor. However, the vast majority of concrete domains are simply not covered by currently available ontologies (or they are not covered to an appropriate level of detail for specialized uses). While we wait for those ontologies to appear, alternative approaches are needed to take advantage of the emerging Semantic Web infrastructure, and to explore new (semantic description-based) methods for tasks like resource browsing and query formulation.

One form of consensus about terminology that is widespread in scientific writing is the use of references to identifiable and available bibliographic sources (i.e. the literature about the subject) in order to put into context a new writing by referring to previous ones. The term literature - defined as "the body of writings on a particular subject" in the Merriam-Webster Collegiate Dictionary - refers to the whole set of writings on a concrete domain, although, due to practical reasons, usually only a subset of them is used in the process of developing a domain ontology about the subject. The identification of this subset is a matter of expert judgement, based on his or her knowledge of the domain, and the reputation of the sources. Obviously, some form of subjectivity can't be completely removed (although it could be reduced, for example, by taking into account bibliometric and related source reputation analysis) from the identification process. Referencing also allows human readers - and, eventually, also to software entities - to be able to discriminate the semantic interpretation of the referenced term between the possible ones, by going to the source/s in which it was defined, possibly traversing several references until reaching the original definition. In this way, a domain ontology can be built on top of a corpus of articles, books and other sources, allowing for the explicit inclusion of several senses for the same term.

Building ontologies that include explicit references to the literature in the domain may constitute an alternative to approaches like socio-cultural Consensus Analysis (Behrens & Kashyap, 2002) that requires the intervention of a group of
representative experts, which can be difficult to set up for building or maintaining a domain ontology. In our view, explicit group processes for developing ontologies may entail a large amount of duplication of effort if they are not preceded by a literature review and synthesis phase, so that first developing a literature-based ontology, which can be done by a single expert, can always be considered a good point of departure.

In this paper, we describe a literature-based approach for the external annotation and browsing of Web resources that are related to a specific domain. The same design philosophy has been used in the so-called review-level database MetaCyc about metabolic pathways, although its ontology only provides a string citations attribute for the specification of literature references (Karp 2000), and, thus, it does not represent literature resources as independent entities. Consequently, it does not support general-purpose encoding of literature annotation utterances such as different word senses or relationships between ontology terms.

The most salient feature of our approach is the explicit inclusion of the articles and books about the domain as "first-class citizens" in the ontology, which are used to annotate and to guide the browsing and searching of the selected resources. In addition, this approach provides a disciplined strategy for annotation and ontology structuring. The approach has already been applied in building two resource browsers. The first was intended as a supplement for teacher education (García & Sicilia, 2001), and the second was built as a repository of information and reports about usability evaluation (García et al, 2002).

We assume that Web resources are persistent, identifiable by an URI, and they could also be used with systems that map identifiers to URIs, like the DOI (Digital Object Identifier) System. This assumption allows for the annotation of any form of Web content (not only HTML pages), provided that its contents do not change over time. Consequently, a change in an annotated Web resource entails that its annotations are no longer valid, and thus they are required to be revised (this can be accomplished by storing the previous revision date for each annotation and comparing it with the Last modified HTTP header).

In addition, the nature of the annotation process excludes personalized pages that provide different contents to different uses (but not dynamically generated pages that are not personalized and possess semantics that do not change over time). Nonetheless, the content fragments that are used to develop adaptive content techniques - as described in (Brusilovsky, 2001) - could be annotated separately and used as an independent resource.

As a result of the application of the literature-based annotation approach, we have developed a methodological blueprint that can be applied to any specific subject in which a commonly agreed body of literature is available, which includes any scientific discipline. The RDF-based software used to browse the annotated resources can be used with no modifications, provided that the same ontology editor is used (minor changes would be needed with other ontology editors, due to slight differences in the RDF mark-up they generate).

The rest of this paper is structured as follows. In the next section, the approach taken to build literature-based ontologies is described, and the following section provides an outline of the method, synthesized from our experience, for their
development. As a proof of concept, the browsing interface built on a specific literature-based ontology is presented, and, in the final section conclusions are drawn.

**Building literature-based-domain ontologies**

The approach we propose to use to build the ontologies includes three different conceptual types of elements, depending on the role the concept plays in the ontology:

- Terms and relations that represent the ontology's domain. Examples of these kinds of terms are usability, user centred design or intelligent interfaces, if we are representing the Human Computer Interaction domain, or Educational Programming Language and Computer-based training programs, if the Learning Technology domain is described.
- Representations of bibliographical resources, that provide information about where and how domain terms and relations between them are defined. These resources must be representative and commonly-accepted by the represented domain community, so that anybody who uses the ontology can identify and consult them.
- The concrete online resources that are annotated. Here we introduce articles or different kinds of resources about the specific domain represented by the ontology, e.g. a paper presented in the ACM CHI or EuroLogo conference.

All these kinds of elements lead to an ontology structure organized in three different levels: **Domain**, **Documentary Sources** and **OnLine Resources**. Besides the type of the terms, this structure has to take into account some meta-information about the defined elements, so that three different layers arise: **Metaclass**, **Class** and **Instance**. Layers are 'transversal', and each one covers several levels (see Figure 1), therefore, they hold different types of information depending on the different level they intersect with. A layer contains the definition of the structure of the elements in the immediately lower layer, and therefore, the set of elements defined in layer i and level j, denoted as M(i,j), are instances of elements included in the set M(i+1,j), for any i between 0 and 2 and any j between 1 and 3.

<table>
<thead>
<tr>
<th>Metaclass Layer</th>
<th>Domain Level</th>
<th>Documentary Sources Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>M(2,2)</td>
<td>M(2,3)</td>
<td></td>
</tr>
<tr>
<td>Class Layer</td>
<td>M(1,1)</td>
<td>M(1,2)</td>
</tr>
<tr>
<td>M(1,3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Instance Layer</td>
<td>M(0,1)</td>
<td>M(0,2)</td>
</tr>
</tbody>
</table>

**Figure 1: Ontology Structure: layers and levels**

The *Metaclass* layer intersects with the *Documentary Sources* level and with the *Domain* level (note that M(2,1) is not filled in Figure 1). In M(2,3), the different kinds of bibliographical sources have been defined. These definitions will be used in the next layer to specify the bibliographical sources that document each concrete domain term or relation. The term Bibliographic-Source can be specialized in Book and Article. An Article can be a Technical-Report, a Journal-Article, an Article-in-a-Book, a Conference-Paper or a
Workshop-Paper. The classes at $M(2,3)$ have been adapted from the (KA)$^2$ ontology (Benjamins & Fensel, 1998). Specifically, we have included all the terms related to Publication, with the exception of not-peer-reviewed Web pages, which we consider to be not commonly-accepted and not recognized by the entire community.

At the Domain level in Metaclass layer, we have defined the kind of ontology elements that can be specified in a knowledge domain. These terms are Domain-Terms and Domain-Relations. Both maintain a relation (Defined-In) with Bibliographic-Source term. All definitions at the Metaclass layer enable the specification of concrete domain classes and concrete documentary sources in the immediately lower one.

The Class layer contains terms of Domain level, Documentary Sources level and Online Resources level. Classes in the Domain level conform a conceptualization of a specific knowledge domain. All terms and relations are instances of the classes Domain-Terms and Domain-Relations defined in Metaclass layer at Domain level, and both are associated to a concrete book or article, which, in turn, will be an instance of a class defined in Metaclass layer at Documentary Sources level (an example of terms and relations in a "Usability Evaluation" domain that are defined in several books and articles is shown in Figure 3). In the Class layer is also necessary to define the kind of online resources that can be annotated with domain terms. We have again used a part of the (KA)$^2$ ontology to specify these resources. In Figure 2 some of the terms extracted and adapted from this ontology are shown as a UML class diagram, according to the knowledge representation described in (Cranefield, 2001). Terms in the Online Resources level are related with the terms in Domain level through different semantic slots, like TopicOf, About, etc.

![Figure 2: On-line resources that can be annotated with domain terms.](image-url)

Terms in $M(2,3)$ (Online Resources level and Class layer) and in $M(1,1)$
(Documentary Sources level and Metaclass layer) could be viewed as the same conceptual items, but we have decided to maintain them as different entities, for two reasons. First, they describe the same kind of element, but neither their intent nor the information requirements put on them are the same. And second, ontology editors that provide metamodelling capabilities such as Protégé require strict metamodelling semantics, that is, the elements at layer $i$ can only be instances of layer $i+1$.

Our approach to resource annotation requires the creation of a new instance of the appropriate kind of resource, which contains the concrete URL of the external resource (an attribute in all classes at $M(1, 1)$). When using this approach, it becomes necessary to select interesting resources and annotate them, creating a browsable resource collection, as we'll describe later. The Instance layer contains the ontology concrete objects, and as shown in Figure 1, it holds instances of terms at the Online Resources and Domain levels. Instances at the former represent annotated resources, and their type is that of the class (in Class layer) from which they are derived. Instances at the latter can be of one of the following kinds:

1. Domain-term specific instances. For example, in a Learning Technologies ontology, "Logo" is an instance of the Programming Language domain term, and if we want to annotate "EuroLogo" as a Conference on "Logo" programming language, we have to create an instance of Conference at $M(0, 1)$, called "EuroLogo", and associate it with the "Logo" term using the relation named About.
2. Instances that represent reified classes, which are needed in some cases to keep the abstraction level in the annotated terms definition, since instances at the Online Resources level maintain associations with one or more instances at the Domain level. For example, if we want to annotate an article about "Computer-Based Training Programmes" in our Learning Technologies ontology, we need to create, besides the specific Article In Journal instance, a reified instance of the class Computer Based Training Programmes to associate both elements, although this one does not represent a specific programme, but the general category also represented in the layer above.

In Figure 3 a partial fragment of a literature-based ontology on Usability Evaluation (a Human-Computer Interaction field) is shown. Terms and relations are separated in the levels and layers described above. Let us describe some facts about this fragment. Usability evaluation can be carried out using by different methods and techniques, as defined in the book "Usability Engineering" by Jakob Nielsen. One of these methods is called Usability Inspection, which is described in "Designing the User Interfaces", by B. Shneiderman. One of the techniques used to inspect usability is the Cognitive Walkthrough, defined in "Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces", by Clayton Lewis at the "ACM International Conference on Human Factors and Computing Systems'90 (CHI90)". Therefore, in the Class layer there are terms such as Usability Evaluation Method, Inspection Method or Cognitive Walkthrough at Domain level, and some instances of Book (Usability Engineering and Usability Inspection Methods), and Conference Paper (for example, that by Clayton Lewis) at the Documentary Sources level.

If we want to annotate a specific online resource, such as Jorgensen's conference paper, "Towards an epistemology of usability evaluation methods", presented at "CybErg 1999", we need to create several instances at the Instance layer. Concretely, at the Online Resources level, CybErg99 must be a Conference instance and Towards an epistemology of usability evaluation methods must be an Article in Conference one. At the Domain level, we need to create a reified instance of Usability Evaluation Methods to
associate the article with its topic. Another example of an annotated online resource is the conference paper "Do Web usability questionnaires measure Web site usability?", presented at the "Conference of the Rocky Mountain Psychological Association 2002". This article examines the psychometric properties of the WAMI questionnaire (a specific questionnaire to measure a Web site's usability), so that, besides the corresponding instances at the Online Resources level, a "normal" instance WAMI of Questionnaire must be created at the Domain level, and both are associated through a relation labelled Study (defined in the Class layer).

In what follows, we shall give examples of how some of the elements in the ontology are translated to XML mark-up.

The following RDF code shows the description of the Inspection Method class, which is a domain term (see rdf:type in the first description) and a subclass of Usability Evaluation Method (see rdfs:subClassOf in the first description). The method is defined, between others, in Shneiderman_98 (see ONTOCHI_01:DOMAIN_DEFINED_IN in the second description), which type is Book (see rdf:type in the second description), an specialization of Bibliographic_Source (see rdfs:subClassOf in the second description).

```xml
<rdf:Description rdf:about="&ONTOCHI_01;Inspection_Method"
    rdfs:label="Inspection Method">
  <rdfs:comment>Methods that use usability specialists, software developers, users and other professionals to examine usability-related aspects of a user interface</rdfs:comment>
  <rdf:type rdf:resource="&ONTOCHI_01;DOMAIN:TERM"/>
  <rdfs:subClassOf rdf:resource="&ONTOCHI_01;Entry_Point"/>
  <ONTOCHI_01:DOMAIN_DEFINED_IN rdf:resource="&ONTOCHI_01;Nielsen_94"/>
  <ONTOCHI_01:DOMAIN_DEFINED_IN rdf:resource="&ONTOCHI_01;Shneiderman_98"/>
  <rdfs:subClassOf rdf:resource="&ONTOCHI_01;Usability_Evaluation_Method"/>
</rdf:Description>
```
To annotate the previously mentioned Jorgensen's conference paper, the following RDF code is required:

```xml
<ONTOCHI_01:Conference_paper rdf:about="&ONTOCHI_01;ontochi_10_00149"
ONTOCHI_01:Author="Jorgensen, A. H."
ONTOCHI_01:Display_Name="Towards an epistemology of usability evaluation methods"
ONTOCHI_01:Title="Towards an epistemology of usability evaluation methods"
ONTOCHI_01:Year="1999"
rdfs:laBel="Towards an epistemology of usability evaluation methods">
<ONTOCHI_01:Author>"Jacobsen, N.E.H."</ONTOCHI_01:Author>
<ONTOCHI_01:Presented_at_conference
rdf:resource="&ONTOCHI_01;ontochi_10_00150"/>
</ONTOCHI_01:Conference_paper>
```

The first element in the above fragment describes a conference paper instance. The first author is A. H. Jorgensen, and the second, N. E. H. Jacobsen. Note that an URI must be specified in the paper instance to enable access to the on-line resource (see `ONTOCHI_01:URI`). The paper was presented at the CybErg conference (described in the second definition), which is internally denoted as `ontochi_10_00150`. The conference is in turn an online resource, so its URI must be specified.

In order to create a relationship between the paper and one of its topics, we have to specify the following RDF code that reifies an instance of Usability Evaluation Methods. In the sentence `ONTOCHI_01:Topic_of` the relation and the related instance are specified:

```xml
<ONTOCHI_01:Usability_Evaluation_Method rdf:about="&ONTOCHI_01;ontochi_11_00157"
ONTOCHI_01:Display_Name="IND_Usability_Evaluation_Method"
ONTOCHI_01:Usability_Evaluation_Method_Name="IND_Usability_Evaluation_Method"
rdfs:laBel="IND_Usability_Evaluation_Method">
<ONTOCHI_01:Topic_of rdf:resource="&ONTOCHI_01;ontochi_10_00149"/>
</ONTOCHI_01:Usability_Evaluation_Method>
```

Towards a method for developing literature-based ontologies

As a result of the application of our literature-based annotation approach, we have developed a preliminary method for the development of resource bases with source annotation. Although we do not claim that it is the ideal method, it has been useful in practical situations, and can be used as a first blueprint for the study, test and further research on more comprehensive methodological frameworks. It essentially consist on four phases, each of them comprised of a number of iterative subtasks, that we have labelled as follows:
1. Source analysis
   a. Source identification
   b. Development of the initial documentary base
   c. Documentary base validation

2. Domain ontology construction
   a. Class hierarchy construction
   b. Class relation elaboration

3. Annotation of a test online resource base

4. Final validation

The Source analysis phase is aimed at producing the computer form of the literature of the domain we are dealing with, i.e. producing the sub-ontology in the class layer at the Documentary Sources level, according to the structure in Figure 1. The first task is the identification of the sources that are considered to be key references in the field (to some extent this is always a matter of opinion, but the same would occur with the decisions taken by an ontogroup). These sources commonly include books, journals and other forms of publications. Although the (relative) importance of the different sources is not explicitly represented in the ontology, it should be used as an input for the second phase. In some cases, citation indexes or other (formal or informal) impact measures (for example, indices in the Research Index search engine) can be used for that purpose. For example, in the case of a Usability Evaluation ontology, the ACM SIGCHI interest group (or other societies in the field) can be considered as an starting point, and the HCI Bibliography site can be used as a source of resources to be encoded.

Once the sources are identified, an initial subset of the bibliographic resources is represented in the ontology. Additional sources will be added in subsequent phases.

Although is virtually impossible to determine whether the sources initially selected are the right set, informal heuristic measures can be used to provisionally validate it. We have used two of these heuristics:

- **Diachronic** analysis, which tries to situate the resources in their historical context. The technique consists simply in sorting the resources by publication date and trying to trace concepts from the most recent to the oldest resources. This enables the identification of terms or classifications that have evolved over time, and those that have become part of the jargon or "tradition" of the field.
- **Compatibility** analysis, which consists of identifying survey-type articles, Web pages or books and matching their reference lists with the selected set of resources. Higher coincidence can be considered an indicator of appropriateness.

The domain ontology construction is aimed at building the Domain level at the Class layer. We have used an approach that focuses first on the classes (terms), and later on class relationships, adopted from the practice of object-oriented analysis methods (Booch, 1993) and (Rumbaugh et al., 1990). In this phase, both the classes and their relationships must be annotated with the sources from which they are extracted (several sources can be used for each of them). Two special relationships must be accounted for in this activity: synonyms and similar or resembling terms and relations. Synonyms are represented at the Domain level by a special relationship of the same name. But in many cases, two classes are not perfectly equivalent, but, to some extent, are similar. In these cases, both terms should be represented independently, and a grade of similarity or proximity - expressed as a value in the [0..1] interval - must be assessed by the ontology.
creator. Similarity is a reflexive, symmetric and transitive, while transitivity is not required for proximity relations. The approach to dealing with those relationships, which is not covered in this paper, is borrowed from the theory of those relations found in Fuzzy Set Theory-related research. See, for example Buckles & Petry, 1982 on similarity relations and Shenoi & Melton, 1989 on proximity relations. The case of a single term with more than one interpretation is modelled by including duplicated entries for the same class, varying their names slightly. This is not a problem in annotating resources, since the description of each class along with its sources and relationships will make clear the underlying concept.

The third phase is intended to validate the domain ontology by actually annotating a representative resource base. Note that the frontier between the third and four phases can be blurred, since we have found that a good approach to ontology building is proceeding from the bottom up by considering a number of sample resources and trying to find the domain classes that best describe them.

Finally, a validation phase must be carried out, in search for flaws, mistakes and deficiencies of the ontology and resource base just constructed. Common ontology evaluation methods like OntoClean (Guarino & Welty, 2002) can be carried out in this phase, and well established design principles are supposed to be followed (Gómez-Pérez, 1999).

The process just described entails a guaranteed minimum quality in the ontology obtained. This way, if the ontology is used as an input for a subsequent ontogroup process, the group of domain experts is provided with a validated "discussion version", boosting the inception phase of its process, and forcing them to give support to criticisms and change proposals in terms of the literature in the field, which would result in updates to the existing ontology. The approach can be considered as a hybrid of inspirational, inductive and synthetic approaches to ontology design according to Holsapple and Joshi (2002), since a single developer may start the process from his own (inspirational) viewpoint, but he has to justify the decisions on existing documentation (in a sort of synthesis), including specific cases in the domain of interest (thus proceeding inductively).

**Browsing and search interfaces**

Searching annotated resources requires novel ways to access information, and the Semantic Web offers the opportunity of defining them (Eberhart, 2001). Annotating resources using ontology terms provides them with semantic information which makes available more precise results in searching, since semantic retrieval instead of complex term-matching is done. We have designed a search engine prototype, called metadataKB, to find resources annotated in the way described in Section 2 (we have not carried out any formal user testing study, but it has already been used by students). Technically, the prototype has been built using Java and it can be used on any Web server that supports the Java Servlet 2.2 specification. The application processes the ontology RDF(s) using JENA 1.1 libraries. At present, the parsed version of the ontology is maintained in memory, concretely, as a data structure attached to a servlet in the application state of the open source Tomcat Web server.

Users do not need to introduce strings to construct a query using the metadataKB. Search criteria are derived from the ontology, since they are built using ontology classes and subclasses, that are shown in the main interface of the
application. This feature enables the use of the search engine independently of a specific ontology or its future updates.

The retrieval process requires the definition of the ontology entry points. Entry points are the most generic meaningful terms in the ontology that enable to enclose the search the first time. On the basis of the entry point, users construct the criterion selecting the most suitable terms. Criterion refinements are also allowed, showing subclasses of the selected terms in the interface. This process can be carried on until no more subclasses can be extracted in the specific hierarchy. So the search criteria are finally composed by the set of classes selected by the user in the application interface.

Search results are obtained in two different ways:

1. Directly recovering those resources annotated with the terms chosen in the prototype interface. This option retrieves all instances of the selected classes. If more than one class is specified, only the instances that belong to all the classes will be retrieved (conjunctive multi-criteria searching).

2. Recovering the semantic relations in which instances of the selected terms take part, in order to offer users the possibility of browsing them. We have denoted as semantic relations those established between two instances (named subject and object) in accordance with the ontology definition.

To obtain the results described, two tasks are carried out:

- First, the intersection of the extensions of the selected classes is computed. We define the extension of a class \( c_i \) as the set containing the instances of class \( c_i \) and instances of all the descendants (subclasses) of \( c_i \). If more than a class is specified as a criterion, only instances that belong to all the classes are retrieved (this makes the set of instances retrieved empty in many cases).

- After that, semantic relations are retrieved. To do that, the union \( U \) of the extensions of the selected classes is computed, and then the subset of relations that incorporate as subject and object instances of \( U \) is retrieved.

To illustrate how the retrieval process must be carried out, an example of a specific search is described. If we want to find conference papers which report usability evaluation surveys using questionnaires, we have to select the terms "Publication" and "Inquiry", shown in the main application interface (both are defined as entry points), as shown in Figure 4. Search criteria can be refined (obviously, if refinement were not done, a huge amount of more generic online resources and semantic relations would be retrieved), so we click on the "Refine" button.

Figure 4. Prototype main window used to search usability evaluation reports.
button ("Refinar" in Spanish). Subclasses of "Article" and "Inquiry" are displayed in the window, and we select "Questionnaire" and "Article" terms. Refinement is done again, and "Conference Paper" and "Questionnaire" are checked (see Figure 5). Once the criteria are adjusted, we click the Search button ("Buscar" in Spanish).

Results are shown in Figure 6. Here we can see that there are no instances of "Conference Paper" and "Questionnaire" at the same time, but different online conference papers related to questionnaires are retrieved. Users can select the most appropriate one on the basis of the semantic information of the relation established between both instances. In our example we are interested in papers that report usability studies carried out with questionnaires, so we can access any of the last three conference papers shown in the interface (using the link at the right side), which make use of PSSUQ, QUIS and an unspecified questionnaire, respectively.

In order to allow users to accurately define the criteria that retrieve the expected results, they are allowed to display at any time the concrete documentary sources that define a term by simply clicking on the link that is associated to its name (see the smaller pop-up window in Figure 4).

**Conclusions and future work**

A literature-based approach has been described that can be used to build ontologies in technical domains that are grounded on a corpus of bibliographical
sources and, therefore, on the evolving consensus of that domain, as reflected in its literature. A three-level and three-layer organization provides a clear separation of concerns between modelling notions, and current RDF-based ontology description languages can be used to encode that structure.

A preliminary sketch of a method for developing literature-based ontologies has been described, that complements existing methods with the specifics of referential term and relation definition.

Finally, software to build resource browsers operating on these ontologies has been built as a proof of concept of its technical feasibility.

Future work will include rewriting the described software libraries to the emerging OWL standard Web ontology description language, and the study of the ontology-based browsers in the broad context of Interactive Information Retrieval (Robins, 2000), taking into account the role of the human user's cognitive processes in term or relation-guided browsing of resources. In addition, the XPath W3C recommendation, which is intended to address parts of XML documents, and, in consequence, of well-formed HTML documents, could be used to annotate fragments inside a Web page. This technique has already been implemented in the Amaya and Mozilla browsers as a result of the Annotea project (Kahan et al., 2002).

Note

1. "RDF": Resource Definition Framework - a framework for the description and exchange of metadata. See, for example, the W3C specification.

References


