Knowledge Media Institute

On the integration of technologies for capturing and navigating knowledge with ontology-driven services

Yannis Kalfoglou[1], John Domingue[1], Leslie Carr[2], Enrico Motta[1],

Maria Vargas-Vera[1], and Simon Buckingham Shum[1]

[1] KMi

[2] Intelligence Agents Multimedia, Dept. of Electronics and CS, University of Southampton, Southampton SO17 1BJ, UK

KMI-TR-106

May 2001

http://www.kmi.open.ac.uk/tr/papers/kmi-tr-106.pdf

On the integration of technologies for capturing and navigating knowledge with ontology-driven services

Yannis Kalfoglou[1], John Domingue[1], Leslie Carr[2], Enrico Motta[1],
Maria Vargas-Vera[1], Simon Buckingham Shum[1]
[1]Knowledge Media Institute(KMi),The Open University,Milton Keynes MK7 6AA, UK
[2]IAM, Dept. of Electronics and CS, University of Southampton, Southampton SO17 1BJ, UK
[1]{y.kalfoglou,j.b.domingue, e.motta, m.vargas-vera, s.buckingham.shum}@open.ac.uk - [2]lac@ecs.soton.ac.uk

ABSTRACT

Nowadays, many distinct communities are researching on technologies for knowledge capturing, modelling, and navigation. Moreover, advances in Internet technology makes it possible to perform most of these tasks on heterogeneous and distributed environments such as the Web. These advances though, have raise the need for knowledge services to accommodate the ever increasing number of Web users. To provide such a service one needs to combine key technologies for different aspects of knowledge management: capturing, modelling, navigating. This should be tightly integrated with the intended service. We describe such an integration effort in this paper. Our domain is a Web-based news repository and we aimed to provide personalised ontology-driven services on the top of it. We used knowledge capturing technologies to populate the underlying ontologies, knowledge modelling techniques to provide reasoning capabilities for the ontology-driven service, and navigating technologies to overlay Web-pages with the ontology-driven service.

Keywords

Knowledge capture, ontology-driven services, knowledge navigation

1. INTRODUCTION

Capturing knowledge contained in diverse and heterogeneously distributed sources has been a longstanding goal of many research efforts, primarily originated from the knowledge acquisition community. Equally important though, is to build structures where this knowledge can be stored and reused whenever appropriate. The knowledge modelling community has been working for years on the uses of ontologies ([21]) to accommodate these tasks. After the knowledge has been captured, stored and represented in appropriate reusable structures, we could reason about it by exploiting the relations that hold between those structures. This provides the ground upon which we could build intelligent services tailored to user preferences. Such a holistic approach in knowledge management, from capturing it to provision of services, is still in its infancy.

The main reason, probably, is the distinct communities who research different aspects of this holistic approach. There is an observed lack of tools that can 'cross' community borders and be used by scholars working on a different aspect of the same problem. As part of our Interdisciplinary Research Collaboration (IRC) we are working on the integration of three major technologies for knowledge management: capturing knowledge by using text engineering methods, modelling knowledge by using ontologies, and navigating knowledge with the use of open hypermedia and annotation of Web documents. An important element of this integration effort is that we apply it to produce an end product, that is, a knowledge service which is Web-accessible and in the case presented here, allow users to personalise a news stories repository.

In this paper, we elaborate on aspects of this integration and the provision of the knowledge service. We do not provide an in-depth analysis of all the issues related with this integration. Rather, we focus on the provision of the knowledge service and we elaborate on the issues that are related to its functionality. This sort of integration is purpose-driven and biased towards the peculiarities of our domain: ontology-driven news services provided to members of an organisation. We have been working on this domain for some time ([16]), with the main focus being the use of ontologies. This integration aims to shed light on the use of complementary technologies, such as capturing and navigating knowledge which could be coupled with knowledge services.

Hence, we start with our efforts on using text engineering techniques to capture knowledge (section 2) which we used to populate the underlying ontology as well as associating it with the knowledge service. Space reasons prevent us from expanding on the knowledge modelling aspect with the use of ontologies. As there is a plethora of conferences and journals dedicated to the particular field, we refer here only to those ontology issues that are related to the provision of the knowledge service. This is described in section 4 whereas in section 3 we elaborate on the integration of Web-based navigational technologies with ontologies and their connection with the knowledge service. We mention related research efforts in section 5 and we conclude the paper in section 6.

2. CAPTURING KNOWLEDGE

Our aim in capturing knowledge is to perform all necessary activities in a non-intrusive manner for the user and yet being able to capture knowledge from a wide variety of resources, ideally Web-accessed documents. In addition, the knowledge we are interested to capture is organisation-specific which makes it possible to build and impose a predefined structure. Indeed, we learned from our experience with the *PlanetOnto* ([5]) suite of tools that organisational

knowledge can be captured and modelled effectively once the purpose and services that will use this knowledge are clearly defined. In our domain, these are news services used by members of an organisation, and we are interested to provide them with news items of their interest.

In this line of work, we are not interested in a full-fledged linguistic analysis of the organisational documents. We have applied though, techniques from Information Extraction (hereafter, IE) field which we coupled with domain-specific templates to identify the parts of a document to be analyzed. The use of templates is purpose-driven as we use the knowledge captured to populate the underlying organisational ontology. Represent organisational knowledge in the form of an ontology allowed us to reason about it, with ontology-driven tools such as the one described in section 4, acting as users' front-ends.

2.1 Template-driven information extraction

In contrast to full linguistic analysis of a document, IE focus only on portions of text that are relevant to a particular domain. From that perspective, IE can be seen as the task of pulling predefined relations from texts as we see in applications of IE in various domains ([17]). Template-driven IE makes it possible to identify relevant information which could be used directly to fulfill a task, like populating an ontology ([22]). If no template applies to the parsed sentence then no information is retrieved.

In our domain of news items describing everyday life in an academic unit we are interested to extract specific information, such names of projects, members of the organisation, funding organisations, awarding bodies, amount of money being awarded, etc. We have followed an event-centred approach in representing the information found in news items ([5]), and we used this event typology as a guide to build templates for extracting information automatically. The template construction and their linking with the IE system is described in detail in [22]. Here we briefly present an example template to illustrate its usefulness in capturing information.

One of the event types in the `KMi Planet` domain is `visiting-a-place-or-people`. For that event we are interested to capture information related to the people or group of people who are visiting, the place they are visiting, the date or duration of their visit, etc. We have define the following template to extract this information:

```
[_,X,_, visited,Y, from, Z,_]
```

This template matches the sentence word list where `X` is recognisable as an entity capable of visiting, `Y` is the place being visited and cannot be a preposition, and `Z` is recognisable as a range of dates by virtue of their syntactic features. The remaining tokens in the sentence are ignored. Each template is triggered by the main verb in any tense. In this template, the trigger word is the verb "visited". In [18] it is argued that linguistic rules could be deployed to help identify trigger words reliably. For example, if the targeted information is the subject or the direct object of a verb then the trigger word should be the main verb. We also make use of the underlying ontology to help us identify proper names for visitors(in case they are KMi members) and whenever this fails we use a named entity recognizer for proper names of people and places.

Assume that we apply this template in a news item which contains the following sentence:

> "AKT collaborating institutions visited Sheffield from January 29-31, 2001 to share ideas and organise the AKT project."

The trigger word 'visited' will activate the template described above and variables `X`, `Y`, and `Z` will be instantiated to visitor, place being visited and range of dates, which will extract the following information:

- visitor: "AKT collaborating institutions"
- place: "Sheffield"
- date: "January 29-31, 2001"

This information will be converted to the appropriate ontology representation language to instantiate it. We do not expand on this topic here as it is peripheral to our issue of capturing information. However, we point the interested reader to [22] where the connection of the IE system with the underlying ontology is described in detail.

2.2 Heuristics-based phrase extraction

An alternative to extract information driven by predefined templates is to scan a document and look for special visual effects in it. As the authors describe in [11], document authors tend to use syntactic methods to delineate key phrases or ideas in documents, such as putting them in italics, identifying them with acronyms and so on. In building a large information retrieval system ([12]), the authors argued for the disadvantages of a full document parsing:

> "[...]very few of the words in a document reflect the underlying meaning and importance of the text, and moreover the distribution of words does not reflect the words or phrases that best characterize the document[...] ideas discussed in a document can often be written in a wide variety of words, which will vary considerably across different authors and different organizations, but the catch-phrases and buzzwords are very often invariant across documents on the same category[...] processing the entire text of a document is extremely costly in computational terms, and can be prohibitive for very large sample sets"

On the contrary, extracting semantically significant phrases is more tractable as it does not require a full parse nor depends on statistical evidence of words occurrences. To implement this idea the use heuristics that capture specific kinds of visual effects have been advocated and applied in [12]. In our work we extended the proposed set of heuristics and apply it for a different purpose: instead of using those heuristics for categorizing documents we tightly integrate them with an ontology-driven service for providing customized news stories to interested users. We present this service in section 4. Initially though, we will describe in detail the set of heuristics we apply for extracting semantically significant phrases and we elaborate on its purpose in the ontology-driven service.

We have choose the following set of heuristics to extract phrases or words that:

- are heavily repeated within the document;
- are fully capitalize(normally, these are acronyms or proper technical names);
- are designated as short phrases (1-5 words) and appear in different format from the surrounding text(e.g. in italics, first letter capitalized);
- are made of compound nouns(3 or more nouns in a row);
- appear in a list of items;
- are section headings;

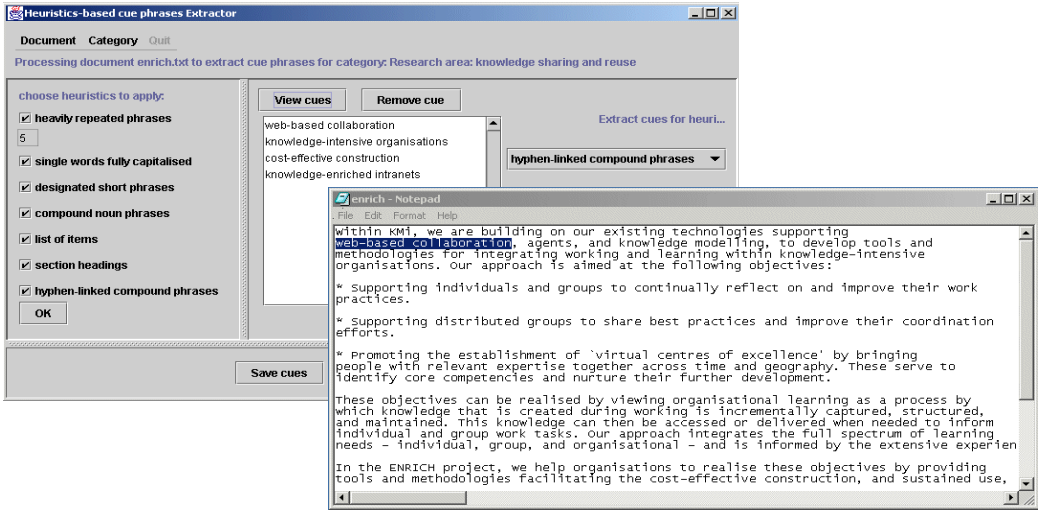

Figure 1: The heuristics-based phrase extraction system.

- are made of hyphen-linked compound phrases.

In figure 1 we include two overlapped screenshots to illustrate the usage of these heuristics. The right window contains the text which we scanned to extract phrases whereas the left window is a screenshot of the heuristics-based extraction tool. As we can see, the user can select any combination of the aforementioned heuristics or apply them all. In this screenshot we see the results of extracting hyphen-linked compound phrases, which are composed of the compound words with a hyphen separating them and the word which follows. For instance, some of the phrases that have been extracted are "web-based collaboration", "knowledge-intensive organizations", etc. These phrases are indeed indicative of the nature of this document which is a description of the KMi project *Enrich* which aims to apply web-based collaboration in knowledge-intensive organizations. Once we extract these phrases, we associate them with ontological categories drawn from KMi's library of ontologies. In that way, we link documents with these categories which helped us in increasing the answer set of related news items in the service we describe in section 4. In this example, the extracted phrases were associated with the category "Research Area: Knowledge Sharing and Reuse". Having identify these phrases in the *Enrich* project document allows us to declare it relevant to the research area "knowledge sharing and reuse".

3. NAVIGATING KNOWLEDGE

KMi Planet uses an ontology to annotate the documents it manages. In this section we present a hypertext linking service which uses the same ontology to apply navigational links to those documents using the organisational knowledge in our domain. Here we focus on the underlying architecture of the link service and how we integrate it with the ontology-driven service.

Our basis was the Conceptual Open Hypermedia Service Environment (hereafter, COHSE[1]), which itself combines ontological reasoning services with an established link service (DLS [2]) to enable documents to be linked together via metadata describing their contents.

In contrast with the common usage of the Web which involves embedding links within documents in the HTML format, open hypermedia systems treat links as first class objects which can be stored and managed separately from the documents in which they are ultimately used.

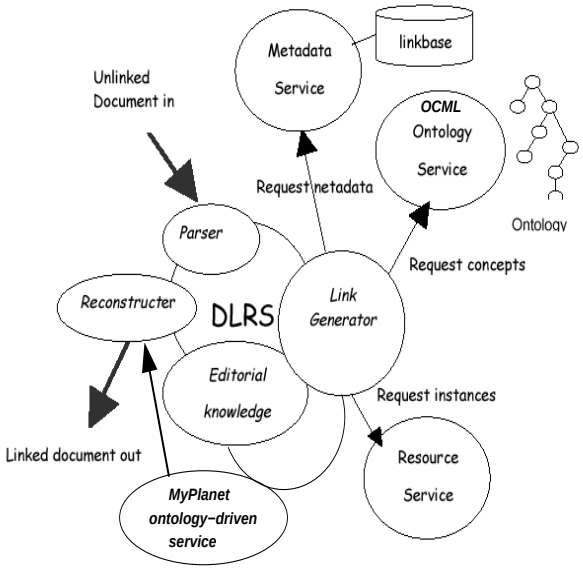

Figure 2: The generic COHSE architecture augmented with an ontology-driven service.

Here the COHSE architecture combines the original link service with a ontological model to enable these independent hypertext links to be targeted on the instances of concepts mentioned in documents, hence "Conceptual Open Hypermedia". Its has four components: (a) an ontology service, (b) a resource service, (c) a metadata service, and (d) a link service (a variation of the original DLS).

The ontology service maintains the ontology and allows the application to interact with it through a well-defined API. It provides operations relating to the content of the conceptual model, e.g. ex-

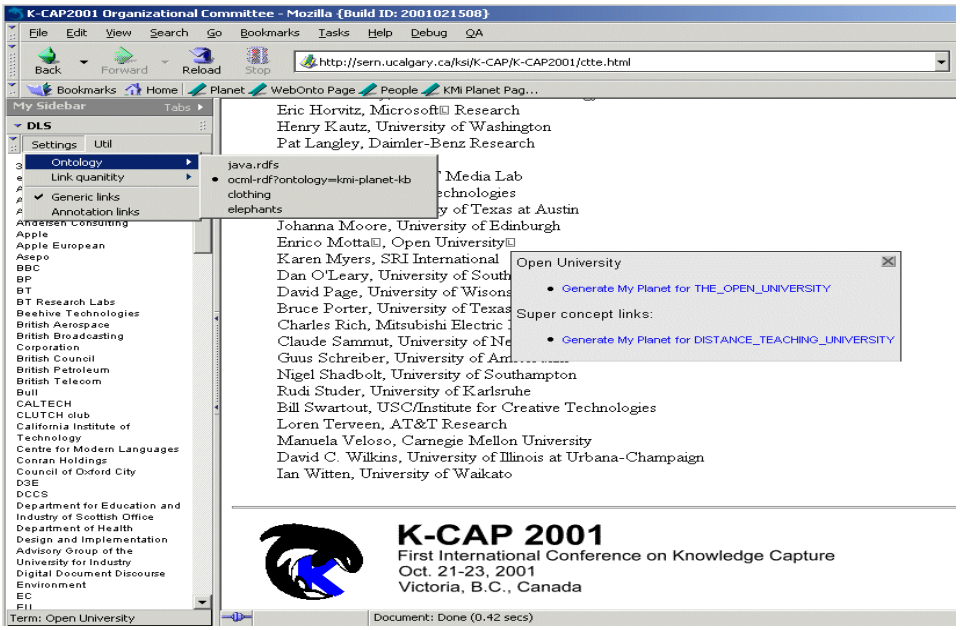

Figure 3: A Web document overlayed with ontology-driven hyperlinks and firing of ontological services.

tend the ontology, query it by returning the parents of children of a concept. The resource service maintains a list of Web pages which represent the various concepts in the ontology. The metadata service allows documents to be decorated with metadata, either words or word combinations in the textual part of the resource itself or could be concepts added explicitly through an external annotation process. The DLS provides presentation of results, dynamically inserting links into Web pages wherever a concept is recognised.

In figure 2 we depict the COHSE architecture diagrammatically. We augmented this architecture with an ontology-driven service. Before we elaborate on this integration we will describe the Link Generator and Editorial Knowledge components of this architecture as both components were used in the integration process. As the authors describe in [1], "the Link Generator module of the DLRS contacts the ontology service to obtain a complete listing of all the language terms that are used to represent the concepts in the ontology. For each of those terms that are recognized as occurring in the document, the generator first asks the ontology service for a preferred term, and then asks for the preferred term to be mapped onto a concept. Having identified a concept from the strings in the document, the link generator contacts the resource service to obtain a list of documents that contain instances of this concept". The Editorial Knowledge module evaluates the number and quality of potential links obtained from the generator. It allows to apply filters, user profiles, similarity matrices etc, and we can hook-in ontology-driven services as we describe in the sequel. The COHSE system is powered by OIL (Ontology Inference Layer). OIL ([7]) unifies the epistemologically rich modelling primitives of frames, the formal semantics and efficient reasoning support of description logics and a mapping to the standard Web metadata language proposals. One of these is RDF (Resource Description Framework).

The COHSE system makes it possible to add various kinds of links to documents in order to create a navigational overlay of Web pages. In addition to this, it provides support for associating certain types of Web-based ontology-driven services with these Web pages. These characteristics were appealing to us for the purpose of our experiment: we were interested to link the personalised ontology-driven service described in the next section[1] with the dynamic linking facilities COHSE provides. Such an integration could lead to a Web page overlayed with specific hyperlinks which could fire personalized services as the one we describe in section 4. In the following paragraphs we describe the integration steps we have made along with an example case.

We had already a rich source of ontologies in our library of Web-accessed (from webonto.ac.uk) knowledge models. We selected the `kmi-planet-kb` which describes the entities that belong to the KMi Planet ([5]) domain. The next step was to make this ontology available to COHSE. As the Link Generator uses OIL to communicate with the underlying ontology we had to write a simple translator to convert our OCML-encoded ontology to RDF format which is supported as a default by OIL. This partial translation enabled the ontology service to pose queries regarding the hierarchical structure of classes in `kmi-planet-kb`. We were not interested in a full translation (i.e., relations, axioms/rules) as this was out of the scope of this experiment. The Link Generator also needs a listing of the language terms that are used to represent concepts in the `kmi-planet-kb`. We imported this as a lexicon written in XML which associates the internal representation of concepts in `kmi-planet-kb` with their string representations. For example, the `org_bbc` concept is associated with strings "BBC" and "British Broadcasting Corporation". Having these two resource files loaded, we then had to invoke the ontology-driven service from within the COHSE system.

We implemented this invocation in the Editorial Knowledge module where the strings found in the document are treated as input for the ontology-driven service. That is because, that service takes ontologically-defined categories as input and returns news items[2]

[1] Originally described in [10].

[2] We coined the term *e-Story* when we refer to this sort of news items as they are stories submitted in the form of an email.

of potential interest after performing ontology-based inferencing. When the service works as a stand-alone tool (as we describe in [10]), these categories are the internal representation of concepts in the `kmi-planet-kb`, for example strings like `org_bbc`. The Editorial Knowledge module, uses the lexicon mentioned before to map the strings found in the document with their internal representations which will then be sent to the ontology-service as input. The output of that service is a Web-page and that made it straightforward to link it with the COHSE system.

Let us go through an example case to demonstrate the usage of this navigational environment. In figure 3 we present a screenshot of a Web page containing the organization committee of the K-CAP 2001 conference. As we can see on the left pane of the Web browser, the COHSE system has already load all the string representations of concepts in the `kmi-planet-kb` ontology. The drop-down menu in that part also shows that we have choose the `kmi-planet-kb` ontology after a translation from OCML to RDF has been made. In the centre of the screenshot we see the annotated Web page. All the strings that belong to the list on the left have been hyperlinked and signposted from the COHSE system with a small graphic widget attached at the end of each string. This is the 'button' which will fire the ontology-driven service that is used in COHSE. In this screenshot we have click on the 'button' at the end of an "Open University" string. This has pop-up a small menu with two choices: (i) generate MyPlanet for "THE_OPEN_UNIVERSITY" and (ii) generate MyPlanet for "DISTANCE_TEACHING_UNIVERSITY". These choices demonstrate the different modes COHSE ontology services can work. We can either call the service directly with the internal representation of the string as the input or we can do some inference before invoking the service. This happened in the second choice where we performed a simple inference step to deduce the super concept of The Open University, that is, distance teaching university, and then we invoked the service with that concept as the input. The underlying `kmi-planet-kb` provided the hierarchical structure.

We close this section by stressing the fact that the link construction software in COHSE can interact with an independent suite of ontology-driven services as we described above. In the following section we will elaborate on the ontology-driven service itself, the MyPlanet news service.

4. KNOWLEDGE SERVICES

Our aim when building *MyPlanet* was to provide a personalised service that would support reasoning and be Web-accessible. We used as a basis the existing infrastructure provided by *PlanetOnto* ([5]), the *KMi Planet* newsletter[3]. *KMi Planet* was originally conceived as an internal newsletter and progressively became an integrated suite of tools for knowledge management. It is used as lightweight communication medium by members of our lab but lacks the advantages of personalised services.

As we describe in [10], we were interested to provide a retrieval method for potential e-Stories of interest that would enable users of *MyPlanet* to read e-Stories that match their preferences instead of browsing the whole archive to find an interesting e-Story. We were also interested to provide support for reasoning when retrieving those e-Stories as opposed to the traditional keywords-based matching algorithms used in conventional search engines.

We devised an interface which allows the user to specify topics of interest (crudely speaking, "the search criteria"). There are two key differences of our approach when comparing it with a search engine: (a) the structure of these topics of interest is ontology-drawn, and (b) the method for matching e-Stories is going beyond the conventional keyword matching in that it is using the association of heuristically extracted phrases with ontological categories described in a previous section (2.2). The advantage of the first difference is that we can reason about the topic being selected by applying ontology-driven deductive heuristics. This would, arguably, give us a more precise answer that matches the topic being selected. In order to work reliably though, we had to increase the set of potential answers. This is where the second key difference came into play: by using the heuristically extracted phrases in our search we achieved a greater recall ratio than we would had with a simple keyword-based match.

The selection of the predefined structure of topic of interest was also an important decision. We had two requirements in mind when selecting the structure: breadth of coverage and depth of reasoning support. The former aims to cover as wide a range of topics as possible whereas the second refers to the underlying ontology structure. Since we had to deal with a well-defined domain, news stories describing events related to everyday matters of an academic organisation, we defined the following structure of topics to *MyPlanet*'s users:

- `Research areas` that are investigated in KMi;
- `Research themes` that are investigated in KMi;
- `Organisations` that KMi collaborates with;
- `Projects` in KMi;
- `Technologies` used in KMi;
- `Application domains` that are investigated in KMi;
- `People` - members of the KMi lab.

These topics are normally encountered in a typical e-Story in the *KMi Planet* stories archive. In addition, it allowed us to satisfy the second requirement, that is depth of reasoning support. Indeed, all of these topics are classes in the underlying KMi Planet ontology and they are related to each other with ontological relations. For example, the following OCML ([15]) code is the definition of an instance of a KMi research and development project, the "KMi Planet" project:

```
(def-instance project-kmi-planet kmi-r&d-project
 ((leading-organization knowledge-media-institute)
  (has-goals "it is designed as a shared work and information space for
  researchers within the OU, at KMi, and at a number of related institutes")
  (has-project-member john-domingue peter-scott)
  (funding-source the-open-university)
  (has-web-address web-page-project-kmi-planet)
  (uses-technology tech-lispweb)
  (associated-products tech-kmi-planet tech-planet-onto)
  (project-application-domain kmi-communication)
  (associated-documents ref-article-kmi-planet-a-web-based-newspaper
   ref-article-A-Knowledge-Based-News-Server-Supporting-Ontology-
   Driven-Story-Enrichment-and-Knowledge-Retrieval)
  (addresses-theme theme-communicating theme-collaborating theme-reasoning)
  (has-research-area res-area-web-based-publishing)))
```

As we can see, this definition is sufficient to support deductive reasoning with respect to the project's research areas, themes, application domain, members, technologies, organisations, all of which are presented as topics of interest to the user of *MyPlanet*. In addition to these simple inference steps, we also use relations to link entities found in news stories, for example people are connected with projects with the following ontological relation:

[3] Accessible online from http://kmi.open.ac.uk/planet/

```
(def-relation involved-in-projects (?x ?project)
    :constraint (and (person ?x)
                     (project ?project))
    :sufficient (or (has-project-member ?project ?x)
                    (has-project-leader ?project ?x)))
```

The OCML language provides support for defining operational options for each relation such as the `:sufficient` construct in our example above. Its purpose is to help characterize the extension of a relation. For the relation given above, it is sufficient to prove that a person is a member or leader of a project in order for the relation `involved-in-project/2` to hold.

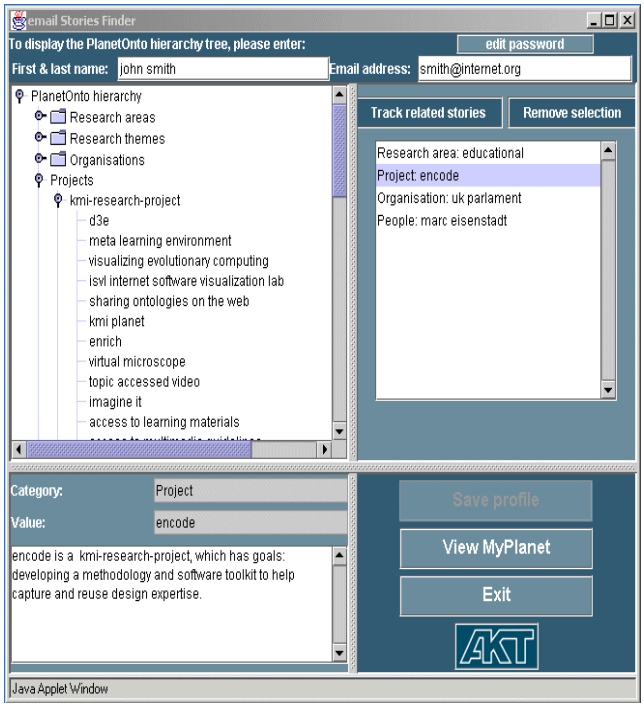

Figure 4: The designated user interface of *MyPlanet*.

We will go through an example case to illustrate the use of *MyPlanet* in a simple e-Story retrieval scenario. In figure 4 we include a screenshot of the designated interface of *MyPlanet*[4]. It is written as a Java Applet which when launched loads all the instances of the classes that compose the structure described above. This will enable users to browse through the available instances and select those for which they want to find e-Stories. In this screenshot, the user 'john smith' has select four instances, of which the 'project: encode' is highlighted. This is the currently viewed item and in the lower left pane we include information related to it, for example a textual description of the project's goals. This information is used to assist to the user select the right topic. We display different types of textual information tailored to the type of category being viewed. For example, when an instance of `People` is viewed then we display the projects that this person is involved to. This information is obtained from the ontology after firing the relevant query.

In the sequel, the user activates the search for related e-Stories by pressing the 'View MyPlanet' button which displays in another browser window the e-Stories found (if any). We include such an e-Story in a screenshot in figure 5. At this point, we deploy association of heuristically extracted phrases with ontological categories to increase the number of related e-Stories. In this screenshot, we have circled such a phrase: "knowledge modelling" which is associated with the homonym ontological category research area. As the project *Encode* has knowledge modelling as research area, this e-Story is said to be related to this project and it is returned as a story of interest although it does not explicitly mention project *Encode*.

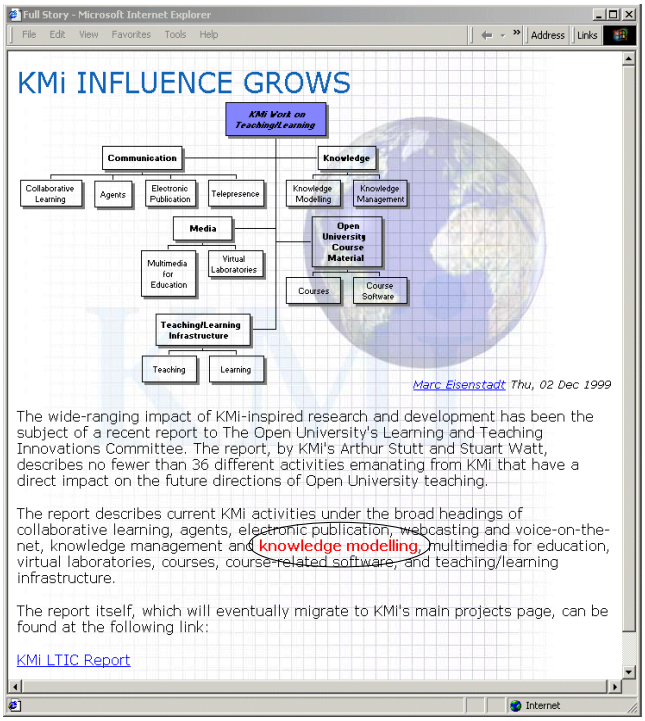

Figure 5: A e-Story returned as relevant to the project *Encode*.

5. RELATED WORK

To the best of our knowledge there are no references in the literature to systems offering the range of services as the one described here. However we have made selective references to related efforts that compared directly with one or more aspects of our system. These are loosely classified as the use of ontologies for providing search services, deploying IE techniques to aid ontology engineering, use of heuristic phrases to learn user interests and alternative approaches to ontological annotation of Web pages.

We first mention two systems that use ontologies to support knowledge-enhanced search. In the *FindUR* project ([14]), the means for search by using ontologies were investigated. McGuinness describes a tool, deployed at the AT&T research labs, which uses ontologies to improve the search experiences from the perspectives of recall and precision as well as ease of query formation. Their tool is mainly targeted to the Information Retrieval research area and aims to improve search engine technology. The idea of deploying ontologies to achieve these goals is similar to our approach which, however, is mostly concerned with using ontologies to structure the search space (i.e., the predefined topics of interest) and increase the

[4]Accessible online from http://eldora.open.ac.uk/my-planet/

answer set. In their work, means for updating the topic sets used to categorize information were investigated. In contrast with our approach where the predefined topics of interests are maintained internally, the *FindUR* team were "experimenting with a collaborative topic-building environment that allows domain experts to expand of modify topic sets directly" [14]. Although this approach has the advantage of speeding up the maintenance task, in our case we see the predefined topics as a stable piece of knowledge over time. On the other hand, the manual maintenance of these topics does not require a great deal of effort as we can use the ontology library to update them whenever this is necessary. We should also point out a similarity in the use of cue phrases and cue words to increase the number of related e-Stories. In the *FindUR* project, the notion of "evidence phrases" was used. However, their definition as "evidence" phrases highlights a difference in their application: as we described in [10], we use heuristically extracted phrases (which we call 'cue phrases') both as abstractions of terms and as evidence, whereas in the *FindUR* domain they used only evidence. For example, as the authors describe, the company *Vocaltec* could be evidence for the topic *Internet telephony* but certainly is not an abstraction of it. In particular, they defined a typology of evidence phrases which were used to increase the number of related answers to a given query. They were deployed in the background along with rules that govern their interrelations. A notable difference though, is that the cue phrases in our system are associated with ontological categories which give us the advantage of exploiting ontology relations to support inferences. In addition, the heuristics-based phrase extraction we described in section 2.2 shows that we could extract them automatically in contrast to *FindUR* where these were edited manually.

A similar approach which deploys *content matching* techniques is described in [8] where the authors present the *OntoSeek* system. It is designed to support content-based access to the Web and its target is the Information Retrieval area with the aim of improving recall and precision. They focussed on specific classes of information repositories: yellow pages and product catalogues. Their underlying mechanism uses conceptual graphs to represent queries and resources descriptions. The similarity of this work with *MyPlanet* lies in the usage of an ontology. However, we deployed our ontology in different phases: in structuring the search space and in increasing the answer set.

IE has emerged as a crucial tool for ontology engineering as it is seen a solution to the ontology learning and scaling up problem. In this direction, [19] and [6] discuss early ideas on the use of IE techniques in order to help them understand complex relationships, statements or terms in semi-structured or unstructured documents. In [13] the authors describe an *ontology learning environment* where a text processing module performs a shallow text analysis to identify linguistically related pairs of words. These are then used as input for a learning algorithm which proposes potential relations between those words. Our work in section 2.1 uses a lighter approach in adapting IE techniques and couples them with domain specific templates as our primary goal is not to learn but rather populate ontologies. In the ontology population area, we should also mention the IMPS (Internet-based Multi-agent Problem Solving) system which uses software agents to conduct knowledge acquisition online by using distributed resources ([3]). One of these agents, *OCA* (Ontology Construction Agent), is used to facilitate the task of constructing an ontology at runtime, that is, querying various resources to fill in the gaps in the ontology. Although the goals of this work were different, the underlying idea for the *OCA* is similar to our efforts of populating the ontology by automatically instantiating classes as we described in section 2.1. Our approach is different in that we deploy IE techniques along with domain specific templates to instantiate specific ontology classes, whereas *OCA* deploys heuristical methods for extraction and focuses on creating an hierarchy lattice of classes of concepts.

When comparing our work of providing personalised services with related efforts, a representative one is the work of [12] on analyzing patterns of access to documents to infer user profiles. Most of these approaches, however, try to induce user interests by employing empirical methods. Although they extract heuristic phrases from users' documents, the difference with *MyPlanet* is that we deliberately impose an ontology-drawn structure of topics of interests, therefore we know the range of 'permissible' user interests beforehand and then we concentrate on how to provide reasoning services related to these interests.

From the open hypermedia perspective we have to mention related efforts in linking hypertext and markup languages with ontologies. Among those, we refer here to SHOE ([9]), Ontobroker ([4]) and OntoAnnotate ([20]). SHOE (Simple HTML Ontology Extension) provides mechanisms that allows Web authors to annotate their documents with semantic information. Annotations are included within pages as mark up using an HTML-based syntax, with a META tag used to inform any agents that the page uses SHOE. In a similar fashion, Ontobroker allows annotation of Web pages with ontological metadata. It provides a more expressive framework for the ontologies, using Frame Logic for the specification of ontologies, annotation data and queries. OntoAnnotate relies on RDF and RDF schema for the annotation of Web pages. The annotation process though is semi-automatic in comparison with the manual annotation of other systems. The COHSE architecture we describe in section 3 differs from these systems in that its purpose is not to support query, but instead to provide extra information and linking for existing Web pages. In addition, it is not confined to a particular set of Web pages(i.e., those who use a specific type of annotation), but any Web page as long as it has mechanisms for recognising the concepts in the documents. As we described in section 3, this could be a simple term matching using a lexicon provided by an ontology, or use of metadata as in the systems mentioned here.

6. SUMMARY AND FUTURE WORK

The integration we achieved in this work was purpose-driven: we had to provide an end product and that led our efforts. In doing so, we were able to be selective on certain issues of the integration. For example, as we described in section 2.1, we were not interested in a full-fledged linguistic analysis of the documents. The use of domain specific templates was adequate for populating the event slots in the underlying ontology. The same holds for the use of heuristically extracted phrases (section 2.2), which were associated with ontological categories in order to support retrieval of related news stories. This is in contrast to conventional heuristic phrase extraction approaches which aim to provide an ad-hoc categorisation of documents. Finally, the nature of the knowledge service, personalisation of a news repository, led us to the particular navigational technology we describe in section 3. A unique characteristic of the COHSE architecture was its seamless integration with *MyPlanet*.

We intend to build on our experiences and extend the current work in three directions. First, the use of IE techniques with domain specific templates worked well for populating particular slots in the underlying ontology. An addition to this, will be to deploy similar templates for learning ontology structures. These could be coupled with more advanced IE techniques, like event recognizers as opposed to named entity recognizers. We also plan to further the use of cue phrases. Although the automation of their extraction is

an important step towards scaling up, we still have to check their 'validity' as cue phrases. That is because, we adopt the view that these phrases should be both an abstraction and evidence. However, most of the automatically extracted phrases qualify as evidence phrases rather than abstractions. At the moment, we manually select those that satisfy the requirements for being cue phrases. Finally, and most importantly, we are working on the provision of more ontology-driven services. The COHSE architecture and its ability to overlay any Web page with ontology-driven services opens new areas of research which we are keen to explore.

Acknowledgements

The research described in this paper is supported by the 6 year Advance Knowledge Technologies Interdisciplinary Research Collaboration - AKT (www.aktors.org) funded by the UK government.

REFERENCES

[1] L. Carr, S. Bechhofer, C. Goble, and W. Hall. Conceptual Linking: Ontology-based Open Hypermedia. In *Proceedings of the 10th International Conference on the World Wide Web(WWW10), Hong Kong*, May 2001.

[2] L. Carr, D. DeRoure, W. Hall, and G. Hill. The Distributed Link Service: A Tool for Publishers, Authors and Readers. *World Wide Web Journal*, 1(1):647–656, 1995.

[3] L. Crow and N. Shadbolt. Acquiring and Structuring Web Content with Knowledge Level Models. In R. Studer and D. Fensel, editors, *Proceedings of the 11th European Workshop on Knowledge Acquisition, Modelling and Management(EKAW'99), Dagstuhl, Germany*, pages 83–101. Springer Verlag, May 1999.

[4] S. Decker, M. Erdmann, D. Fensel, and R. Studer. Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In R & et.al. Meersman, editor, *Proceedings of DS-8, Semantic Issues in Multimedia Systems, Boston, MA, USA*, pages 351–369, 1999.

[5] J. Domingue and E. Motta. Planet-Onto: From News Publishing to Integrated Knowledge Management Support. *IEEE Intelligent Systems*, 15(3):26–32, 2000.

[6] A. Faatz, T. Kaamps, and R. Steinmetz. Background knowledge, indexing and matching interdependencies if document management and ontology maintenance. *position paper in Proceedings of the ECAI2000 workshop on Ontology Learning(OL2000), Berlin, Germany*, August 2000.

[7] D. Fensel, I. Horrocks, F. van Harmelen, S. Decker, M. Erdmann, and M. Klein. OIL in a Nutshell. In *Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management(EKAW'00), Juan-Les-Pins, France*, October 2000.

[8] N. Guarino, C. Masolo, and G. Vetere. OntoSeek: Content-Based Access to the Web. *IEEE Intelligent Systems*, 14(3):70–80, May 1999.

[9] J. Heflin and J. Hendler. Dynamic ontologies on the Web. In *Proceedings of the 17th National Conference on Artificial Intelligence(AAAI'00), Austin, TX, USA*, August 2000.

[10] Y. Kalfoglou, J. Domingue, E. Motta, M. Vargas-Vera, and S. Buckingham-Shum. MyPlanet: an ontology-driven Web-based personalised news service. In *Proceedings of the IJCAI'01 workshop on Ontologies and Information Sharing, Seattle, USA*, August 2001.

[11] B. Krulwich. Learning document category descriptions through the extraction of semantically significant phrases. In *Proceedings of the IJCAI'95 Workshop on Data Engineering for Inductive Learning*, 1995.

[12] B. Krulwich and C. Burkley. The InfoFinder Agent: Learning User Interests through Heuristic Phrase Extraction. *IEEE Intelligent Systems*, 12(5):22–27, 1997.

[13] A. Maedche and S. Staab. Semi-automatic engineering of ontologies from texts. In *Proceedings of the 12th International Conference on Software Engineering and Knowledge Engineering(SEKE2000), Chicago, IL, USA*, pages 231–239, July 2000.

[14] L.D. McGuinness. Ontological Issues for Knowledge-Enhanced Search. In N. Guarino, editor, *Proceedings of the 1st International Conference on Formal Ontology in Information Systems(FOIS'98), Trento, Italy*, pages 302–316. IOS Press, June 1998.

[15] E. Motta. *Reusable Components for Knowledge Models: Case Studies in Parametric Design Problem Solving*, volume 53 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 1999. ISBN: 1-58603-003-5.

[16] E. Motta, S. Buckingham-Shum, and J. Domingue. Ontology-driven document enrichment: principles, tools and applications. *International Journal of Human-Computer Studies*, (52):1071–1109, 2000.

[17] D. Proux and Y. Chenevoy. Natural language processing for book storage: Automatic extraction of information from bibliographic notices. In *Proceedings of the Natural Language Processing Pacific Rim Symposium(NLPRS'97)*, pages 229–234, 1997.

[18] E. Riloff. An empirical study of automated dictionary construction for information extraction in three domains. *AI Journal*, (85):101–134, 1996.

[19] C. Roux, D. Proux, F. Rechermann, and L. Julliard. An ontology enrichment method for a pragmatic information extraction system gathering data on genetic interactions. *position paper in Proceedings of the ECAI2000 Workshop on Ontology Learning(OL2000), Berlin, Germany*. August 2000.

[20] S. Staab, A. Maedche, and S. Handschuh. An annotation framework for the Semantic Web. In *Proceedings of the 1st International Workshop on MultiMedia Annotation(MMA2001), Tokyo, Japan*, January 2001.

[21] M. Uschold and M. Gruninger. Ontologies: principles, methods and applications. *The Knowledge Engineering Review*, 11(2):93–136, November 1996.

[22] M. Vargas-Vera, J. Domingue, Y. Kalfoglou, E. Motta, and S. Buckingham-Shum. Template-driven information extraction for populating ontologies. In *Proceedings of the IJCAI'01 Workshop on Ontology Learning, Seattle, WA, USA*, August 2001.