

Towards a Knowledge-Aware Office Environment

Leslie Carr, Timothy Miles-Board, Gary Wills,
Arouna Woukeu, and Wendy Hall

Intelligence, Agents, Multimedia Group
University of Southampton, Southampton, UK
`{lac,tmb,gbw,aw1,wh}@ecs.soton.ac.uk`

Abstract. The objective of the Semantic Web is to make the Web amenable to computer processing, and hence to improve the value that humans can obtain from it. One of the oft-touted user benefits is improved searching: better-described resources allow search engines to provide better-targeted search results. The aim of this paper is to investigate the way in which Semantic Web technologies can be applied to an office environment as the context in which people work and carry out day-to-day document tasks, focusing on the issues of creating and re-using knowledge-rich documents within that environment. To address these issues, we have analysed a business writing scenario and integrated an established commercial off-the-shelf office production environment with knowledge-aware services to assist the author in carrying out writing tasks within that scenario.

1 Introduction

Organisations have become increasingly concerned with knowledge management [1], amassing large intranets and multimedia information Web sites in order to capture their corporate knowledge [2] and use it to inform future discussions, decisions and activities. To put this into effect, organisations require a suitably sophisticated IT strategy and infrastructure, however without planning and support these knowledge-bases may be constructed and used in an unsystematic fashion which becomes impractical to manage as the intranet grows in size and complexity. Web and hypermedia design methods enable Web site designers to provide effective navigational access to information resources by cataloguing the kinds of available information; document management systems further help by managing metadata and providing classification and querying support to locate relevant information.

Beyond these dissemination facilities, the ways in which information is put to use varies with the role of each user within the organisation and also with the type and context of the information. To create new material, an author must be familiar with the context in which he or she is writing. Gaining this understanding from an intranet or Web site may come through browsing (navigation through link following) or searching (navigation through content matching). However, staff do not often have sufficient time for unbounded browsing and searching to

evaluate the precise relevance of a large range of supplementary and supporting documentation. Take for example the situation where a manager is writing a policy statement; it is necessary to draw together information held in a number of business documents: corporate vision statements, corporate strategy documents, departmental policy documents, management summaries, financial reports and public relations statements *etc.*

While reading the content of those documents, the manager will also want to know their purpose (e.g. the intended audience) and authorship (e.g. the author's position of influence) in order to be confident about any judgements made about the documents. What they could reasonably ask of a semantically enriched support environment is to identify relevant material from appropriate documents, based on the context in which new material is being written. The new document should then be published in a form that facilitates reuse of the new knowledge embodied within it, and which provides explicit references to the sources of any reused knowledge, so that (for example) the connection between an institution's three critical success factors and the three section headings in the middle of its corporate strategy document can be identified.

The Semantic Web [3] provides the basis of just such an infrastructure by augmenting documents with explicit statements of semantics, allowing the Web (or an intranet) to be used for more than a human-browseable repository of information. The meaning of the published documents, knowledge about their authors and the reasons for their publication can all be made explicit (to one degree or another) and so be used to infer contextually appropriate knowledge. This paper reports the latest efforts of the Writing in the Context of Knowledge (WiCK) project¹ in investigating the use of Semantic Web technologies in a business-type environment, where authors create and re-use knowledge-rich documents. After considering related work that contributes to our understanding of how this union of technologies can be achieved (Sect. 2), we outline our approach to supporting the specific task of writing a funding proposal (Sect. 3), and finally introduce WiCKOffice, our knowledge-aware office environment which leverages knowledge-aware services to assist an author in such a scenario (Sects. 4 and 5).

2 Related Work

When writing new documents, trying to find and reuse the intrinsic knowledge held in other sources amongst an ever-rising mountain of material is at best impractical. Numerous efforts have contributed to our understanding of how this situation can be improved. ARIA [4], for example, supports email or web page authoring based on a semantically annotated photo database. By continuously monitoring the text typed by the author against a domain ontology, ARIA recommends photos from the database that seem appropriate to illustrate the various facets of the unfolding narrative. CREAM [5] helps the writer produce the text itself, by dragging and dropping knowledge fragments from an ontology

¹ <http://wick.ecs.soton.ac.uk/>

browser into a text editor — for example a dropped slot inserts a text rendering of the slot value (with a link back to the source).

The potential research and commercial benefits of bringing these knowledge-aware processes into the office arena have not gone unnoticed. Microsoft Word, for example, is the most massively adopted product for authoring text documents [6]; authors can therefore adopt new knowledge-aware extensions without learning a new production environment and without sacrificing familiar features [7]. SemanticWord [6], a Microsoft Word-based environment, adds several toolbars to the standard interface which support the creation of semantic annotations in documents and templates according to selected ontologies (local or imported from the Semantic Web). Annotations are “carried over” in text cut/copy and paste operations, facilitating a level of knowledge reuse between documents. SemanticWord also offers a more proactive annotation feature which the author experiences through the Microsoft Smart Tags interface: as the author types the text content of the document, it is processed by an information extraction component which relates instances and values appearing in the text to ontology instances and types, visually highlighting the matched text in the document. Through the Smart Tags “action” menu, the author can examine the highlighted entities and convert them into semantic annotations.

Although provoking a range of reactions upon its release [8], Smart Tag technology has also been adopted by other office-based initiatives, including SemTalk [9] and OntoOffice [10]. As with SemanticWord, recognised concepts and instances are highlighted with Smart Tags. However, the kinds of action offered differs between systems: in SemTalk, for example, the author can access and edit the underlying ontological model; in OntoOffice, a search for context-relevant documents can be initiated.

Going beyond simply supporting writing in the context of an underlying ontology, the WiCK project has attempted to build on these initiatives by considering an office environment in which several knowledge-bases and knowledge-aware services exist and actively assist the author by providing targeted knowledge that would otherwise need to be searched for both manually and individually.

3 Scenario: Writing Funding Proposals

The task of writing a funding proposal is common in industrial and commercial environments; here, we consider a hypothetical funding proposal for a research project in an academic environment. The proposal is directed at the UK’s Engineering and Physical Sciences Research Council (EPSRC), which has a well-defined procedure for submitting, reviewing, and selecting proposals for funding, and provides a standard form² (the Je-SRP1) and a comprehensive guidance document³ on how to fill out the form, create the supplementary documentation, and submit it for consideration.

² <http://www.epsrc.ac.uk/website/commonpages/downloads.aspx?CID=4482>

³ <http://www.epsrc.ac.uk/website/commonpages/downloads.aspx?CID=8621>

The Je-SRP1 form itself serves as an administrative summary of the research proposal, collecting together the relevant information about the hosting organisation, project investigators, project partners (for joint proposals), referees, staff (including visiting researchers), and travel and equipment costs. The ‘meat’ of the proposal is contained in the supplementary document — the Case for Support — the composition of which is tightly defined in the guidance notes. The rules for the Case define the formatting (constraints on page length, font sizes *etc.*), the information content, and the structure of parts and sections where each of these pieces of information should be placed. The content of the Case includes previous research track records, proposed research programme and methodology, proposed dissemination routes, and justifications for each of the resources requested in the Je-SRP1 form.

4 Our Approach

In order to properly model the Je-SRP1 form and Case for Support document and the knowledge they contain, and hence be able to deploy it usefully in a computational environment more complex than a search engine, our scenario requires a number of ontologies. Firstly, we need to understand and model *what is being written about*. To meet the requirements for our scenario, we propose a *research ontology* to describe the stakeholders and activities who participate in research — the researchers, their publications, research interests, conferences and journals, and a *subject ontology* to describe the area in which we wish to conduct research, the problems that we wish to address and the methods, systems and approaches which have been described in the literature.

Having modelled the subject domain of the writing task, we next need to understand the ‘design specification’ for the writing task itself — *what needs to be written*. We therefore propose a *document ontology* to make explicit the semantic structure of the proposal documents — the pages, sections, paragraphs, forms, and fields. In order to explicitly model the *type* of information that the author must enter into each part of this structure, a *project ontology* capturing the activity of undertaking work — the ideas of work package, budget, personnel, milestones *etc.* — and a *proposal ontology* — describing the objectives, beneficiaries, funding call, and programme of activity for the project are proposed.

It follows that filling in the Je-SRP1 form is mainly a matter of choosing appropriate instances against the above ontologies from the knowledge-base. The initial fields on the form are for the host organisation’s name and reference, and the name of the Principal Investigator. The explicit constraints on this information (according to the guidance notes) are that the host organisation must be of a specific type (e.g. UK Higher Education) and that the PI must be employed by the host organisation and must have a contract of the appropriate type (an academic, duration at least as long as the project lifetime). These simple constraints can easily be modelled as verification conditions on data entry, or as queries upon the knowledge-base to select an appropriate list of choices. De-

constructing the form in this way therefore provides an outline proposal ontology, with the guidance notes document supplying the constraints.

Creating the Case for Support document is more involved, as the author is required to construct a text, rather than enter data into clearly labelled spaces on a form. However the guidance notes indicate very clearly the kind of information that is expected in each part of the document. Examining the bullet points which give instructions for Part 1 of the Case for Support, we can see what basic information is required from the knowledge-base, in addition to the kind of processing and analysis which would need to be performed on it by computational services in order to provide valuable assistance to the author.

Provide a summary of the results and conclusions of recent work in the technological/scientific area which is covered by the research proposal. Include reference to both EPSRC funded work and non-EPSRC funded work. Details of relevant past collaborative work with industry and/or with other beneficiaries should be given.... This specifies a literature review; the knowledge is described by the subject and research ontologies. A simple query of the knowledge-base would provide a list of potentially relevant papers, but a more advanced reasoning agent would be required in order to assist the author in evaluating the relative significance of the projects and papers.

Part 2 of the Case for Support requires a different kind of knowledge support, for instance within the *Program and Methodology* section: *Identify the overall aims of the project and the individual measurable objectives against which you would wish the outcome of the work to be assessed.* This information does not exist in the knowledge-base; it is invented as an integral part of the creation of a new research undertaking. However, authors may be assisted by seeing the aims and objectives of similar, recent or successful project proposals, especially if they do not have much experience of proposal writing to draw on. In other words a lack of personal experience could be supplemented by directed browsing of an institutional memory.

This brief examination of the EPSRC guidance notes shows how heavily the writing process (both apparently free-text content creation and information recall) is constrained and specified by the appropriate ontologies, opening the possibility of substantive help from a suitably equipped knowledge environment.

4.1 Proposed Architecture

Figure 1 illustrates our proposed knowledge-aware office environment, WiCKOffice, designed in response to the opportunities for functionality identified in the previous section. In this environment, knowledge is managed by two knowledge-bases, both based on the AKT 3Store platform [11]. The AKT knowledge-base models the UK Higher Education computer science community⁴ (expressed using the AKT Reference Ontology⁵), and hence provides a suitable research ontology for our purposes. A WiCK knowledge-base hosts the additional ontologies.

⁴ <http://www.hyphen.info/>

⁵ <http://www.aktors.org/publications/ontology/>

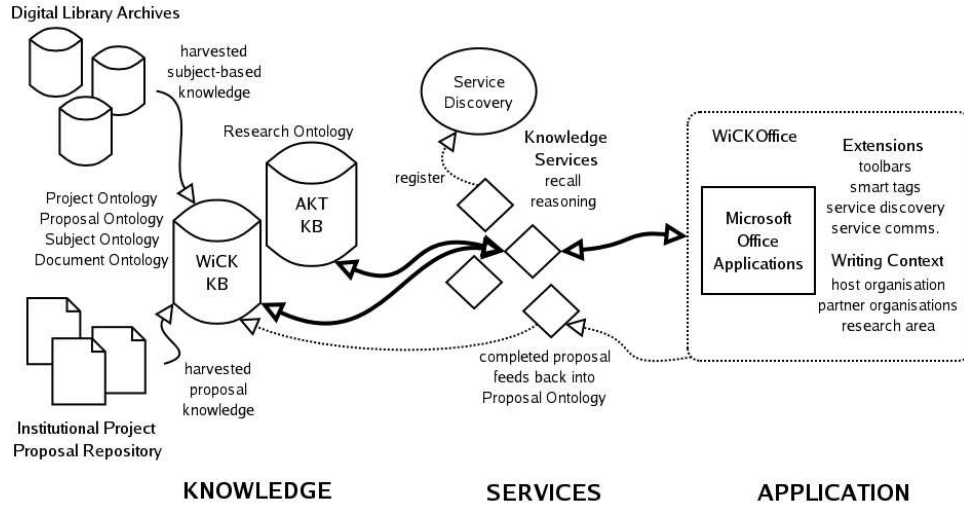


Fig. 1. The proposed WiCKOffice knowledge writing environment.

Instances for the proposal ontology are acquired from previous EPSRC project proposals; we envision Semantic Web agents trawling digital library archives and automatically constructing and populating the subject ontology. WiCK extensions to the Microsoft Office environment utilise key computational knowledge services to assist the writing task, and to update the knowledge-bases when the writing task is completed (for example, new proposals becoming part of the “institutional memory”).

5 WiCKOffice: A Knowledge Writing Environment

Based on the opportunities for functionality identified in the previous section, our modelling and development efforts to date have produced a coherent WiCKOffice environment in which several knowledge services are available to authors. A *knowledge fill-in* service and *knowledge recall* service are motivated by the need to provide timely and convenient access to knowledge, which would otherwise have to be manually ‘looked up’ on the institutional intranet. A third service, *in-line guidelines*, also assists recall by exposing guidelines and constraints captured from the design specification (the EPSRC guidance notes) that are relevant to the part of the proposal document currently being worked on, presenting them to the user via the Microsoft Office Assistant interface (Fig. 2).

5.1 Filling in Forms

The *knowledge fill-in* service assists the author in filling in the Je-SRP1 form. For example, the author can specify the (partial) name of the Principal Investigator

B. Investigators Please give details of each investigator using the same format as below.

| Details | Principal Investigator |
|--|------------------------|
| Title | |
| Forename(s) | |
| Surname | |
| Organisation | |
| Division or Department | |
| Post will outlast project (Y/N) | |
| % time committed to project | |
| Other commitments (description and average hours per week) | |

Investigators

- The Principal Investigator (PI) should normally be the individual who is leading the management of the project. He/she will be the Council's main contact for the proposal. The PI may be supported by a number of Co-Investigators.
- All Investigators must meet the Council's eligibility requirements for research grants – see below and check the relevant Council's funding policy booklet for further details and contacts concerning eligibility.
- Other requirements are that:
 - all fields must be completed for each Investigator
 - one of the Investigators must be from the Host Organisation (i.e. the organisation at which the grant would be held)

OK




Fig. 2. In-line guidelines presented via the Microsoft Office assistant.

and instruct the service to retrieve appropriate (in context) instances from the knowledge-base to automatically fill in the remainder of the required information.

The majority of the information required to provide an assisted knowledge fill-in service for the Je-SRP1 form is already provided by the AKT Reference Ontology (the *research* ontology in our scenario). However, leveraging this service is not as simple as filling each part of the form with an appropriate instance selected from the research ontology — different parts of the Je-SRP1 form “share” data about the same concept. For example, information relating to the Principal Investigator must entered in three different locations: section 1B (page 1) requires the PI’s title, name, organisation, department, and commitments to other projects; section 2B (page 12) requires the PI’s name (for the proposal declaration); and section 3B (page 13) requires the PI’s contact telephone number, email address, fax number, *etc.*

We have therefore used Microsoft Office 2003’s new “smart documents” feature to add *semantic structure* to the otherwise unstructured Je-SRP1 template in the form of an XML Schema derived from the document ontology. The XML Schema identifies each ‘sub-form’ of the Je-SRP1 and groups together related sub-forms (thus, for example, describing the fact that information about the PI is shared by sub-forms 1B, 2B, and 3B). Each individual form field is marked up with three attributes — the ID of the sub-form to which the field belongs, a boolean value indicating whether that field is a preferred search field (in the case of the Je-SRP1, the PI’s first name and surname are good search terms for a person instance in the research ontology; knowing the PI’s title may not so

helpful), and finally a **filled-in-by** attribute which identifies the slot of the matching knowledge instance which should be used to actually provide a value for the field.

When the author partially fills in a sub-form (Fig. 3a) and presses the “Fill-In” button, the XML structure of the document is consulted to determine which fields are part of the current sub-form (and also which fields are part of other sub-forms that share data with the current sub-form). Fields in the current sub-form with an **is-search-field** attribute value of **true** are then used by the knowledge fill-in service to construct an RDQL query to extract matches from the research ontology. In the case that multiple instances match the query, these instances are presented to the author who chooses the appropriate match. Finally, the **filled-in-by** attribute is used to map the slot values of the returned instance to each associated field (Fig. 3b).

Document2 - Microsoft Word

File Edit View Insert Format Tools Window Help

EPSRC Engineering & Physical Sciences Research Council
Polaris House, North Star Avenue, Swindon, Wiltshire,
United Kingdom SN2 1ET
Telephone +44 (0) 1793 444000
Web http://www.epsrc.ac.uk/

COMPLIANCE WITH THE DATA PROTECTION ACT 1998
In accordance with the Data Protection Act 1998, the personal data provided on this form will be processed by EPSRC, computerised database and/or manual files. Further details may be found in the **guidance notes**

EPSRC Reference:

RESEARCH PROPOSAL

1 DETAILS OF PROPOSAL

You should read the separate notes for guidance and the 'EP' completing any research proposal. Form Je-SRP1 (EPSRC) in case for support. EPSRC will reject incomplete research prop

A. Organisation Where Grant Would Be Held

Organisation: RO Reference:
Division or Department:

B. Investigators Please give details of each investigator below. Please provide the details of any additional investigators using the same format as below.

| Details | Principal Investigator (PI) <input type="button" value="FILL IN"/> | Co-Investigator 1 <input type="button" value="FILL IN"/> |
|--------------|--|--|
| Title | | |
| Forename(s) | | |
| Surname | Carr | |
| Organisation | | |

a. Author fills in partial details.

B. Investigators Please give details of each investigator below. Please provide the c using the same format as below.

| Details | Principal Investigator (PI) <input type="button" value="FILL IN"/> |
|------------------------|--|
| Title | Dr |
| Forename(s) | Les |
| Surname | Carr |
| Organisation | University of Southampton |
| Division or Department | Intelligence, Agents, Multimedia Group |

b. All sub-forms sharing data with current sub-form are populated from matching instance.

Fig. 3. Using the knowledge fill-in service to help complete the Je-SRP1 form.

Recently, the EPSRC rolled out its own assisted form filling system, the Je-S1 e-form⁶, which provides some equivalent functionality to this service. Provided that each party has previously registered their details with the system, the author can select the host organisation, principal and co-investigators, referees and other staff from checklists and then download a partially completed JE-SRP1 form which contains all the required details of the selected parties, but still requires some unaided ‘mandraulic’ effort to complete in full. By contrast, we argue that the WiCKOffice approach of leveraging the functionality of multiple services operating over diverse *knowledge* sources (including, but not restricted to, employee data and information harvested from personal webpages and directories) not only allows authors to be aided in filling in *all* aspects of the Je-SRP1 form but also potentially offers wider applicability (adding new types of form requires only that form’s semantic structure be elicited according the document ontology) than a *data*-based application. By explicitly modelling the knowledge in proposal forms, we can also reasonably ask questions such as “which proposals has *Jane Doe* been involved in?” of an institution’s document repository.

5.2 Knowledge in the Right Place at the Right Time

The *knowledge recall* service assists the author in quickly and conveniently recalling appropriate knowledge from the research environment. Example (contextual) queries include “what papers relevant to this proposal have been published recently?”, or “what relevant projects has this person worked on?”. In response to such queries, appropriate knowledge from the knowledge-bases is selected and inserted directly into the document in the form of ‘potted’ summaries.

As with the *knowledge fill-in* service, the AKT Reference Ontology provides the majority of knowledge utilised by this service. In the current implementation, given the name of a recognised person, project or place, the knowledge recall service assists the writer in recalling facts about it. We have seen that recent incarnations of Microsoft Office already provide a mechanism for recognising terms and presenting available “actions” associated with that term to the user in the form of Smart Tags. However, in the case of the Case for Support document, the author’s information requirements depend on the section or part of the document currently being worked on. For example, the author might expect that typing “Les Carr” in the *Previous Research* section would make available options to “auto-summarise” or browse those facets of Les Carr’s previous research history most relevant to the current proposal, whereas typing “Les Carr” in the *References* section would make available options to insert Les Carr’s most recent and relevant publications, and typing “Les Carr” in the *Researcher Curriculum Vitae* section would make available options to insert a “mini CV” with information appropriate to the proposal (with links to knowledge sources in each case). However, prior to the release of Microsoft Office 2003, the actions made available through Smart Tags have been static; Office 2003 allows the set of avail-

⁶ <https://je-s.rcuk.ac.uk/>

able actions to be determined dynamically when the author activates (clicks on) a Smart Tag [13].

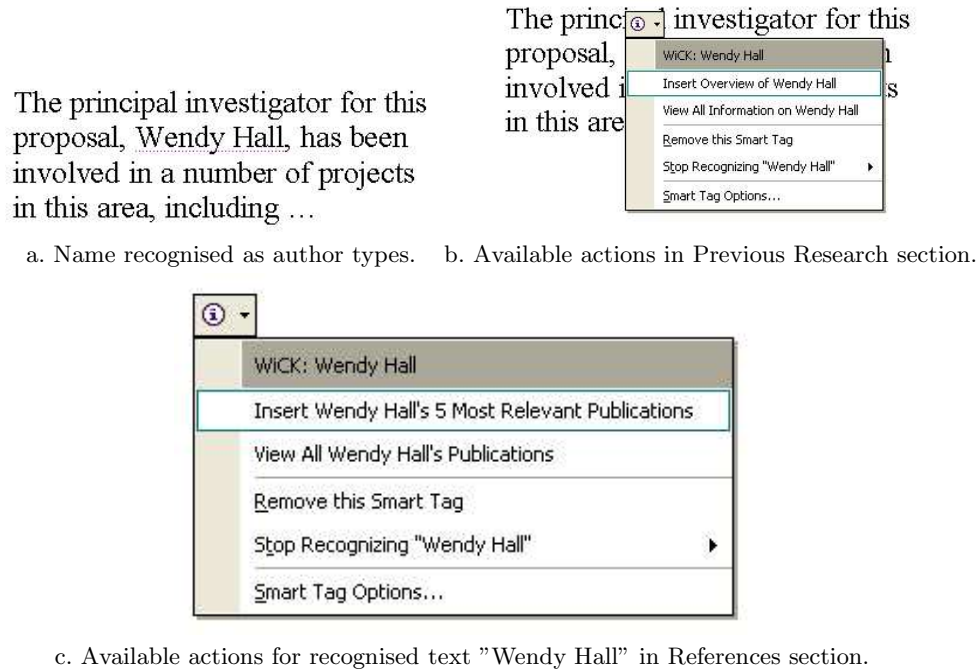


Fig. 4. Using the knowledge recall service, via the WiCKOffice Smart Tag.

An XML Schema derived from the document ontology is again used to make explicit the structural semantics of the Case for Support document. When the author activates a WiCK Smart Tag by clicking on a highlighted term in the text, the XML structure of the document is consulted to work out which part of the document the text appears in (e.g. Background, References) and the actions offered by available services which are appropriate to the type of knowledge required in that section are presented (Fig. 4). We therefore describe this service as providing knowledge in the right *place* (i.e. appropriate to the author's current location in the document) at the right *time* (when a name of a recognised person, place or project is typed by the author).

5.3 Planned Future Services

Two further knowledge-based services are currently under development within the project proposal writing scenario. Using an appropriate proposal ontology, an *augmented experience* service provides the author with access to the "institutional memory" of previous research proposals, thereby augmenting the author's

own experience of proposal writing (“what works? what doesn’t work?”). For example, the author is assisted in evaluating the most important beneficiaries of the proposed research by being shown the beneficiaries put forward by other proposals (with an indication as to whether those proposals were subsequently approved or otherwise).

An *assisted writing* service attempts to assist the author in making higher-level decisions about relevant content to include in the proposal by suggesting appropriate instances from the subject ontology (for example, relevant projects, papers, resources) based on both the writing context and the text that the author has already written. For example, this service uses an internal bibliometric reasoning engine to detect that although the author has referred to a number of *knowledge acquisition*-related projects in the *Background* section⁷ of the Case for Support, one statistically significant project has not yet been mentioned, and so offers to create a summary of the project from the relevant instances in the knowledge-base (gathering details of key personnel and publications) and inserts the information into the appropriate sections of the Case document.

6 Conclusions and Future Work

Semantic Web technologies are beginning to build large, flexible knowledge stores which can be leveraged for diverse purposes within an organisation. This paper has reported the latest efforts of a project to assist authors in creating and re-using knowledge-rich documents within such an environment. This paper has three contributions. First we have analysed the knowledge flow in an business writing task, specifically the creation of a project proposal aimed at a funding body, in this case a research council. Secondly we have identified that this task can be constrained and specified by five ontologies: a *research ontology* (stakeholders and activities participating in research) and *subject ontology* (the area in which we wish to conduct research) capture *what we are writing about*; a *document ontology* (proposal structure), *project ontology* (the activity of undertaking work) and *proposal ontology* (objectives, beneficiaries, funding call *etc.*) model *what needs to be written*. Finally, we have used the above to integrate an office environment with knowledge-aware services to demonstrate that with a suitable set of ontologies and a supportive knowledge-aware environment, an author can be assisted in producing knowledge-rich documents.

Our future work plans, aside from continued implementation of our integrated office environment, include a more detailed focus on the processes and mechanisms by which the knowledge provided by the AKT and WiCK knowledge-bases can be updated and maintained as more and more research proposals are produced. We also plan to carry out a systematic user evaluation — academics who write research proposals as part of their day-to-day work are a readily exploitable human resource in our department. Lastly, we are also working on a

⁷ Guidance notes: “Demonstrate a knowledge and understanding of past and current work in the subject area both in the UK and abroad.”

writing methodology for creating more complex, knowledge-rich documents such as multi-faceted Web sites and hypertexts.

7 Acknowledgements

This work has been funded in part by the EPSRC Knowledge Writing in Context (KWiC) project (GR/R91021/01) — now known as *Writing in the Context of Knowledge (WiCK)* — and the EPSRC Advanced Knowledge Technologies IRC (GR/N15764/01) in the UK.

References

1. Shadbolt, N.R., Milton, N.: From Knowledge Engineering to Knowledge Management. *British Journal of Management* **10** (1999) 309–322
2. Heath, I., Wills, G., Crowder, R., Hall, W., Ballantyne, J.: A New Authoring Methodology for Large-Scale Hypermedia Applications. *Multimedia Tools and Applications* **12** (2000) 129–144
3. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. *Scientific American* (2001)
4. Lieberman, H., Liu, H.: Adaptive Linking between Text and Photos Using Common Sense Reasoning. In: *Proceedings of the Conference on Adaptive Hypermedia and Adaptive Web Systems*, Malaga, Spain. (2002) 2–11
5. Handschuh, S., Staab, S.: Authoring and Annotation of Web Pages in CREAM. In: *Proceedings of the Eleventh International World Wide Web Conference*, Honolulu, Hawaii, USA. (2002)
6. Tallis, M.: Semantic Word Processing for Content Authors. In: *Proceedings of the Knowledge Markup & Semantic Annotation Workshop*, Florida, USA. (2003) Part of the Second International Conference on Knowledge Capture, K-CAP 2003.
7. Tallis, M., Goldman, N.M., Balzer, R.M.: The Briefing Associate: Easing Authors into the Semantic Web. *IEEE Intelligent Systems* **17** (2002)
8. Hughes, G., Carr, L.: Microsoft Smart Tags: Support, ignore or condemn them? In: *Proceedings of the ACM Hypertext 2002 Conference*, Maryland, USA. (2002) 80–81
9. Fillies, C., Wood-Albrecht, G., Weichardt, F.: A Pragmatic Application of the Semantic Web using SemTalk. In: *Proceedings of the Eleventh International World Wide Web Conference*, Honolulu, Hawaii, USA. (2002) 686–692
10. ontoprise GmbH: OntoOffice Tutorial. http://www.ontoprise.de/documents/tutorial_ontooffice.pdf (2003)
11. Harris, S., Gibbins, N.: 3store: Efficient Bulk RDF Storage. In: *Proceedings of the 1st International Workshop on Practical and Scalable Semantic Systems (PSSS'03)*, Sanibel Island, Florida. (2003) 1–15
12. Fillies, C.: On Visualizing the Semantic Web in MS Office. In: *Proceedings of the 6th International Conference on Information Visualisation (IV'02)*, London, England. (2002) 441–446
13. Kunicki, C.: What's New with Smart Tags in Office 2003. *MSDN OfficeTalk* (2003) Available from <http://msdn.microsoft.com/library/en-us/dnofftalk/html/office01022003.asp>.