Advanced Knowledge Technologies at the Midterm: Tools and Methods for the Semantic Web

Nigel Shadbolt[1,2], Fabio Ciravegna[3], John Domingue[4], Wendy Hall[2], Enrico Motta[4], Kieron O'Hara[2], David Robertson[5], Derek Sleeman[6], Austin Tate[5], Yorick Wilks[3]

1. Introduction

In a celebrated essay on the new electronic media, Marshall McLuhan wrote in 1962:

> Our private senses are not closed systems but are endlessly translated into each other in that experience which we call consciousness. Our extended senses, tools, technologies, through the ages, have been closed systems incapable of interplay or collective awareness. Now, in the electric age, the very instantaneous nature of co-existence among our technological instruments has created a crisis quite new in human history. Our extended faculties and senses now constitute a single field of experience which demands that they become collectively conscious. Our technologies, like our private senses, now demand an interplay and ratio that makes *rational* co-existence possible. As long as our technologies were as slow as the wheel or the alphabet or money, the fact that they were separate, closed systems was socially and psychically supportable. This is not true now when sight and sound and movement are simultaneous and global in extent. (McLuhan 1962, p.5, emphasis in original)

Over forty years later, the seamless interplay that McLuhan demanded between our technologies is still barely visible. McLuhan's predictions of the spread, and increased importance, of electronic media have of course been borne out, and the worlds of business, science and knowledge storage and transfer have been revolutionised. Yet the integration of electronic systems as open systems remains in its infancy.

Advanced Knowledge Technologies (AKT) aims to address this problem, to create a view of knowledge and its management across its lifecycle, to research and create the services and technologies that such unification will require. Half way through its six-year span, the results are beginning to come through, and this paper will explore some of the services, technologies and methodologies that have been developed. We hope to give a sense in this paper of the potential for the next three years, to discuss the insights and lessons learnt in the first phase of the project, to articulate the challenges and issues that remain.

The WWW provided the original context that made the AKT approach to knowledge management (KM) possible. AKT was initially proposed in 1999, it brought together an interdisciplinary consortium with the technological breadth and complementarity to

---

[1] Authorship of the Scientific Report has been a collaborative endeavour with all members of AKT having contributed.

[2] University of Southampton.
[3] University of Sheffield.
[4] The Open University.
[5] University of Edinburgh.
[6] University of Aberdeen.

create the conditions for a unified approach to knowledge across its lifecycle. The combination of this expertise, and the time and space afforded the consortium by the IRC structure, suggested the opportunity for a concerted effort to develop an approach to advanced knowledge technologies, based on the WWW as a basic infrastructure.

The technological context of AKT altered for the better in the short period between the development of the proposal and the beginning of the project itself with the development of the semantic web (SW), which foresaw much more intelligent manipulation and querying of knowledge. The opportunities that the SW provided for e.g., more intelligent retrieval, put AKT in the centre of information technology innovation and knowledge management services; the AKT skill set would clearly be central for the exploitation of those opportunities.

The SW, as an extension of the WWW, provides an interesting set of constraints to the knowledge management services AKT tries to provide. As a medium for the semantically-informed coordination of information, it has suggested a number of ways in which the objectives of AKT can be achieved, most obviously through the provision of knowledge management services delivered over the web as opposed to the creation and provision of technologies to manage knowledge.

AKT is working on the assumption that many web services will be developed and provided for users. The KM problem in the near future will be one of deciding which services are needed and of coordinating them. Many of these services will be largely or entirely legacies of the WWW, and so the capabilities of the services will vary. As well as providing useful KM services in their own right, AKT will be aiming to exploit this opportunity, by reasoning over services, brokering between them, and providing essential meta-services for SW knowledge service management.

Ontologies will be a crucial tool for the SW. The AKT consortium brings a lot of expertise on ontologies together, and ontologies were always going to be a key part of the strategy. All kinds of knowledge sharing and transfer activities will be mediated by ontologies, and ontology management will be an important enabling task. Different applications will need to cope with inconsistent ontologies, or with the problems that will follow the automatic creation of ontologies (e.g. merging of pre-existing ontologies to create a third). Ontology mapping, and the elimination of conflicts of reference, will be important tasks. All of these issues are discussed along with our proposed technologies.

Similarly, specifications of tasks will be used for the deployment of knowledge services over the SW, but in general it cannot be expected that in the medium term there will be standards for task (or service) specifications. The brokering meta-services that are envisaged will have to deal with this heterogeneity.

The emerging picture of the SW is one of great opportunity but it will not be a well-ordered, certain or consistent environment. It will comprise many repositories of legacy data, outdated and inconsistent stores, and requirements for common understandings across divergent formalisms. There is clearly a role for standards to play to bring much of this context together; AKT is playing a significant role in these efforts. But standards take time to emerge, they take political power to enforce, and they have been known to stifle innovation (in the short term). AKT is keen to understand the balance between principled inference and statistical processing of web content. Logical inference on the Web is tough. Complex queries using traditional AI inference methods bring most distributed computer systems to their knees. Do we set

up semantically well-behaved areas of the Web? Is any part of the Web in which semantic hygiene prevails interesting enough to reason in? These and many other questions need to be addressed if we are to provide effective knowledge technologies for our content on the web.

2. AKT knowledge lifecycle: the challenges

Since AKT is concerned with providing the tools and services for managing knowledge throughout its lifecycle, it is essential that it has a model of that lifecycle. The aim of the AKT knowledge lifecycle is not to provide, as most lifecycle models are intended to do, a template for knowledge management task planning. Rather, the original conceptualisation of the AKT knowledge lifecycle was to understand what the difficulties and challenges there are for managing knowledge whether in corporations or within or across repositories.

The AKT conceptualisation of the knowledge lifecycle comprises six challenges, those of acquiring, modelling, reusing, retrieving, publishing and maintaining knowledge (O'Hara 2002, pp.38-43). The six challenge approach does not come with formal definitions and standards of correct application; rather the aim is to classify the functions of AKT services and technologies in a straightforward manner.

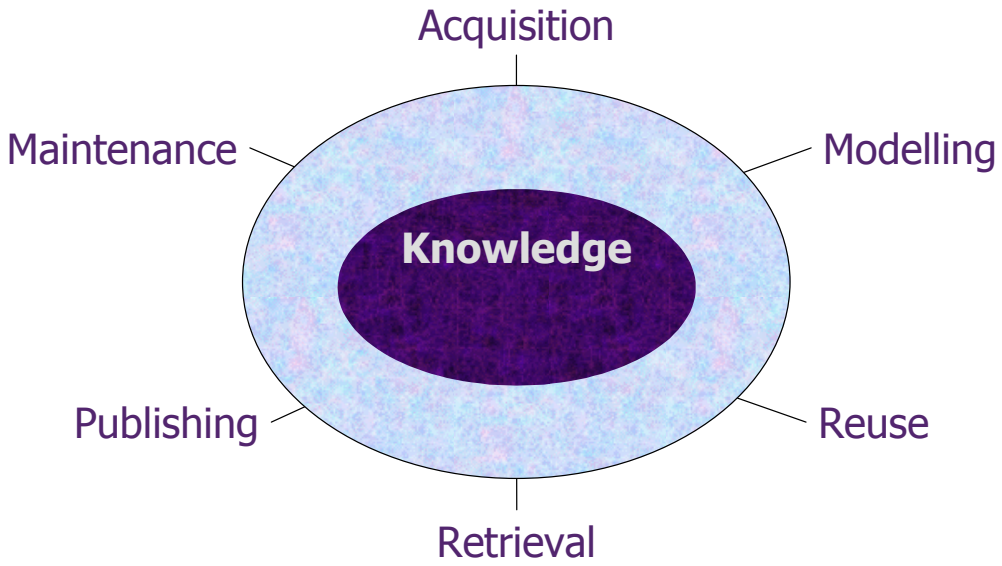


Figure 1: AKT's six knowledge challenges

This paper will examine AKT's current thinking on these challenges. An orthogonal challenge, when KM is conceived in this way (indeed, whenever KM is conceived as a series of stages) is to integrate the approach within some infrastructure. Therefore the discussion in this paper will consider the challenges in turn (sections 3-8), followed by integration and infrastructure (section 9). We will then see the AKT approach in action, as applications are examined (section 10). Theoretical considerations (section 11) and future work (section 12) conclude the review.

3. Acquisition

Traditionally, in knowledge engineering, knowledge acquisition (KA) has been regarded as a bottleneck (Shadbolt & Burton, 1990). The SW has exacerbated this bottleneck problem; it will depend for its efficacy on the creation of a vast amount of

annotation and metadata for documents and content, much of which will have to be created automatically or semi-automatically, and much of which will have to be created for legacy documents by people who are not those documents' authors.

KA is not only the science of extracting information from the environment, but rather of finding a mapping from the environment to concepts described in the appropriate modelling formalism. Hence, the importance of this for acquisition is that – in a way that was not true during the development of the field of KA in the 1970s and 80s – KA is now focused strongly around the acquisition of ontologies. This trend is discernable in the evolution of methodologies for knowledge intensive modelling (Schreiber et al, 2000).

Therefore, in the context of the SW, an important aspect of KA is the acquisition of knowledge to build and populate ontologies, and furthermore to maintain and adapt ontologies to allow their reuse, or to extend their useful lives. Particular problems include the development and maintenance of large ontologies, creating and maintaining ontologies by exploiting the most common, but relatively intractable, source of natural language texts. However, the development of ontologies is also something that can inform KA, by providing templates for acquisition.

AKT has a number of approaches to the KA bottleneck, and in a paper of this size it is necessary to be selective (this will be the case for all the challenges). In this section, we will chiefly discuss the harvesting and capture of large scale content from web pages and other resources, (section 3.1), content extraction of ontologies from text (section 3.2), and the extraction of knowledge from text (section 3.3). These approaches constitute the AKT response to the new challenges posed by the SW; however, AKT has not neglected other, older KA issues. A more traditional, expert-oriented KA tool approach, will be discussed in section 3.4.

3.1. Harvesting

AKT includes in its objectives the investigation of technologies to process a variety of knowledge on a web scale. There are currently insufficient resources marked up with meta-content in machine-readable form. In the short to medium term we cannot see such resources becoming available. One of the important objectives is to have up to date information, and so the ability to regularly harvest, capture and update content is fundamental. There has been a range of activities to support large-scale harvesting of content.

3.1.1 Early harvesting

Scripts were written to "screen scrape" university web sites (the leading CS research departments were chosen), using a new tool Dome (Leonard & Glaser 2001), that is an output of the research of an EPSRC student.

Dome is a programmable XML/HTML editor. Users load in a page from the target site and record a sequence of editing operations to extract the desired information. This sequence can then be replayed automatically on the rest of the site's pages. If irregularities in the pages are discovered during this process, the program can be paused and amended to cope with the new input.

We see below (Figure 2) the system running, and processing a personal web page, also shown. A Dome program has been recorded which removes all unnecessary

elements from the source of this page, leaving just the desired data, and the element names and layout have been changed to the desired output format, RDF.

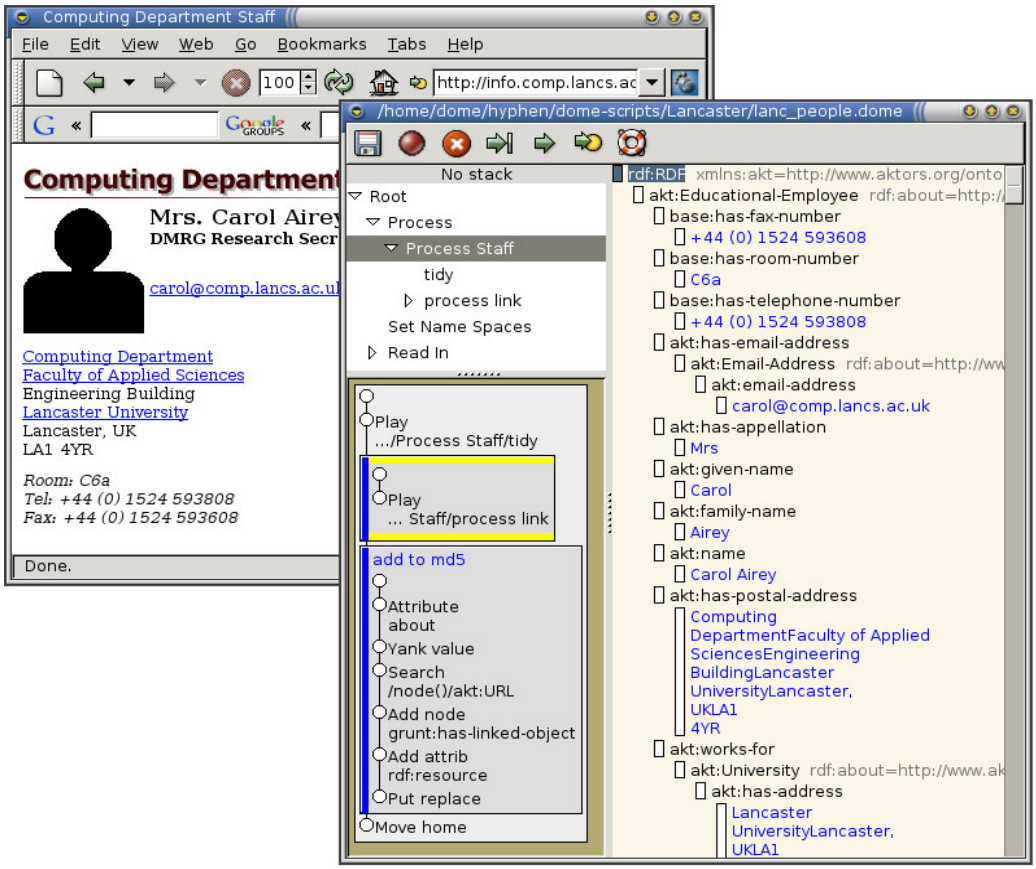


Figure 2: A Dome Script to produce RDF from a Web Page

Other scripts have been written using appropriate standard programming tools to harvest data from other sources. These scripts are run on a nightly basis to ensure that the information we glean is as up to date as possible. As the harvesting has progressed, it has also been done by direct access to databases, where possible. In addition, other sites are beginning to provide RDF to us directly, as planned.

The theory behind this process is that of a bootstrap. Initially, AKT harvests from the web without involving the personnel at the sources at all. (This also finesses any problems of Data Protection, since all information is publicly available.) Once the benefits to the sources of having their information harvested becomes clear, some will contact us to cooperate. The cooperation can take various forms, such as sending us the data or RDF, or making the website more accessible, but the preferred solution is for them to publish the data on their website on a nightly basis in RDF (according to our ontology). These techniques are best suited to data which is well-structured (such as university and agency websites), and especially that which is generated from an underlying database.

As part of the harvesting activity, and as a service to the community, the data was put in almost raw form on a website registered for the purpose: www.hyphen.info. Figure 3 shows a snapshot of the range of data we were able to make available in this form.

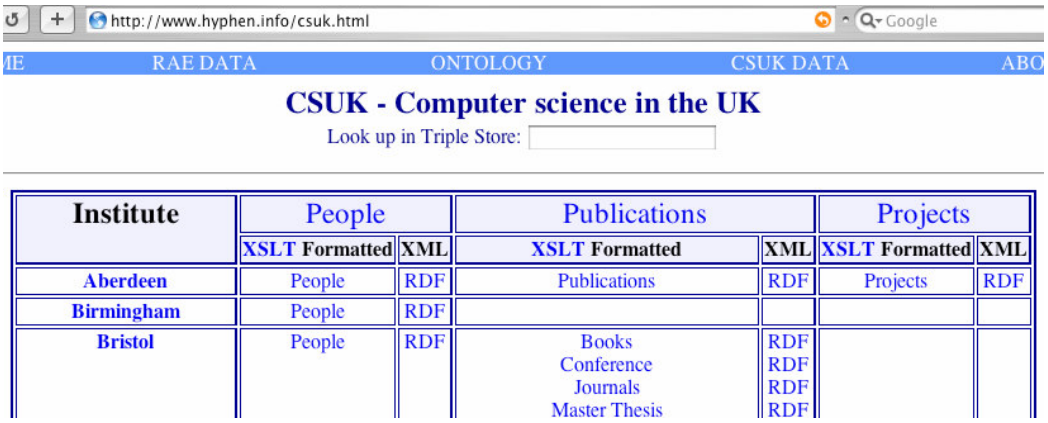


Figure 3: www.hyphen.info CS UK Page

3.1.2 Late harvesting

The techniques above will continue to be used for suitable data sources. A knowledge mining system to extract information from several sources automatically has also been built (Armadillo – cf section 6.2.2), exploiting the redundancy found on the Internet, apparent in the presence of multiple citations of the same facts in superficially different formats. This redundancy can be exploited to bootstrap the annotation process needed for IE, thus enabling production of machine-readable content for the SW. For example, the fact that a system knows the name of an author can be used to identify a number of other author names using resources present on the Internet, instead of using rule-based or statistical applications, or hand-built gazetteers. By combining a multiplicity of information sources, internal and external to the system, texts can be annotated with a high degree of accuracy with minimal or no manual intervention. Armadillo utilizes multiple strategies (Named Entity Recognition, external databases, existing gazetteers, various information extraction engines such as Amilcare – section 6.1.1 – and Annie) to model a domain by connecting different entities and objects.

3.2. Extracting ontologies from text: Adaptiva

Existing ontology construction methodologies involve high levels of expertise in the domain and the encoding process. While a great deal of effort is going into the planning of how to use ontologies, much less has been achieved with respect to automating their construction. We need a feasible computational process to effect knowledge capture.

The tradition in ontology construction is that it is an entirely manual process. There are large teams of editors or, so-called, ‘knowledge managers’ who are occupied in editing knowledge bases for eventual use by a wider community in their organisation. The process of knowledge capture or ontology construction involves three major steps: first, the construction of a concept hierarchy; secondly, the labeling of relations between concepts, and thirdly, the association of content with each node in the ontology (Brewster et al 2001a).

In the past a number of researchers have proposed methods for creating conceptual hierarchies or taxonomies of terms by processing texts. The work has sought to apply methods from Information Retrieval (term distribution in documents) and Information Theory (mutual information) (Brewster 2002). It is relatively easy to show that two terms are associated in some manner or to some degree of strength. It is possible also

to group terms into hierarchical structures of varying degree of coherence. However, the most significant challenge is to be able to label the nature of the relationship between the terms.

This has led to the development of Adaptiva (Brewster et al 2001b), an ontology building environment which implements a user-centred approach to the process of ontology learning. It is based on using multiple strategies to construct an ontology, reducing human effort by using adaptive information extraction. Adaptiva is a Technology Integration Experiment (TIE – section 3.1 of the Management Report).

The ontology learning process starts with the provision of a seed ontology, which is either imported to the system, or provided manually by the user. A seed may consist of just two concepts and one relationship. The terms used to denote concepts in the ontology are used to retrieve the first set of examples in the corpus. The sentences are then presented to the user to decide whether they are positive or negative examples of the ontological relation under consideration.

In Adaptiva, we have integrated Amilcare (discussed in greater detailed below in section 6.1.1). Amilcare is a tool for adaptive Information Extraction (IE) from text designed for supporting active annotation of documents for Knowledge Management (KM). It performs IE by enriching texts with XML annotations. The outcome of the validation process is used by Amilcare, functioning as a pattern learner. Once the learning process is completed, the induced patterns are applied to an unseen corpus and new examples are returned for further validation by the user. This iterative process may continue until the user is satisfied that a high proportion of exemplars is correctly classified automatically by the system.

Using Amilcare, positive and negative examples are transformed into a training corpus where XML annotations are used to identify the occurrence of relations in positive examples. The learner is then launched and patterns are induced and generalised. After testing, the best, most generic, patterns are retained and are then applied to the unseen corpus to retrieve other examples. From Amilcare's point of view the task of ontology learning is transformed into a task of text annotation: the examples are transformed into annotations and annotations are used to learn how to reproduce such annotations.

Experiments are under way to evaluate the effectiveness of this approach. Various factors such as size and composition of the corpus have been considered. Some experiments indicate that, because domain specific corpora take the shared ontology as background knowledge, it is only by going beyond the corpus that adequate explicit information can be identified for the acquisition of the relevant knowledge (Brewster et al. 2003). Using the principles underlying the Armadillo technology (cf. Section 6.2.2), a model has been proposed for a web-service, which will identify relevant knowledge sources outside the specific domain corpus thereby compensating for the lack of explicit specification of the domain knowledge.

3.3. KA from text: Artequakt

Given the amount of content on the web there is every likelihood that in some domains the knowledge that we might want to acquire is out there. Annotations on the SW could facilitate acquiring such knowledge, but annotations are rare and in the near future will probably not be rich or detailed enough to support the capture of extended amounts of integrated content. In the Artequakt work we have developed tools able to search and extract specific knowledge from the Web, guided by an

ontology that details what type of knowledge to harvest. Artequakt is an Integrated Feasibility Demonstrator (IFD) that combines expertise and resources from three projects – Artiste, the Equator and AKT IRCs.

Many information extraction (IE) systems rely on predefined templates and pattern-based extraction rules or machine learning techniques in order to identify and extract entities within text documents. Ontologies can provide domain knowledge in the form of concepts and relationships. Linking ontologies to IE systems could provide richer knowledge guidance about what information to extract, the types of relationships to look for, and how to present the extracted information. We discuss IE in more detail in section 6.1.

There exist many IE systems that enable the recognition of entities within documents (e.g. 'Renoir' is a 'Person', '25 Feb 1841' is a 'Date'). However, such information is sometimes insufficient without acquiring the relation between these entities (e.g. 'Renoir' was born on '25 Feb 1841'). Extracting such relations automatically is difficult, but crucial to complete the acquisition of knowledge fragments and ontology population.

When analysing documents and extracting information, it is inevitable that duplicated and contradictory information will be extracted. Handling such information is challenging for automatic extraction and ontology population approaches.

Artequakt (Alani et al 2003b, Kim et al 2002) implements a system that searches the Web and extracts knowledge about artists, based on an ontology describing that domain. This knowledge is stored in a knowledge base to be used for automatically producing tailored biographies of artists.

Artequakt's architecture (Figure 4) comprises of three key areas. The first concerns the knowledge extraction tools used to extract factual information items from documents and pass them to the ontology server. The second key area is the information management and storage. The information is stored by the ontology server and consolidated into a knowledge base that can be queried via an inference engine. The final area is the narrative generation. The Artequakt server takes requests from a reader via a simple Web interface. The reader request will include an artist and the style of biography to be generated (chronology, summary, fact sheet, etc.). The server uses story templates to render a narrative from the information stored in the knowledge base using a combination of original text fragments and natural language generation.

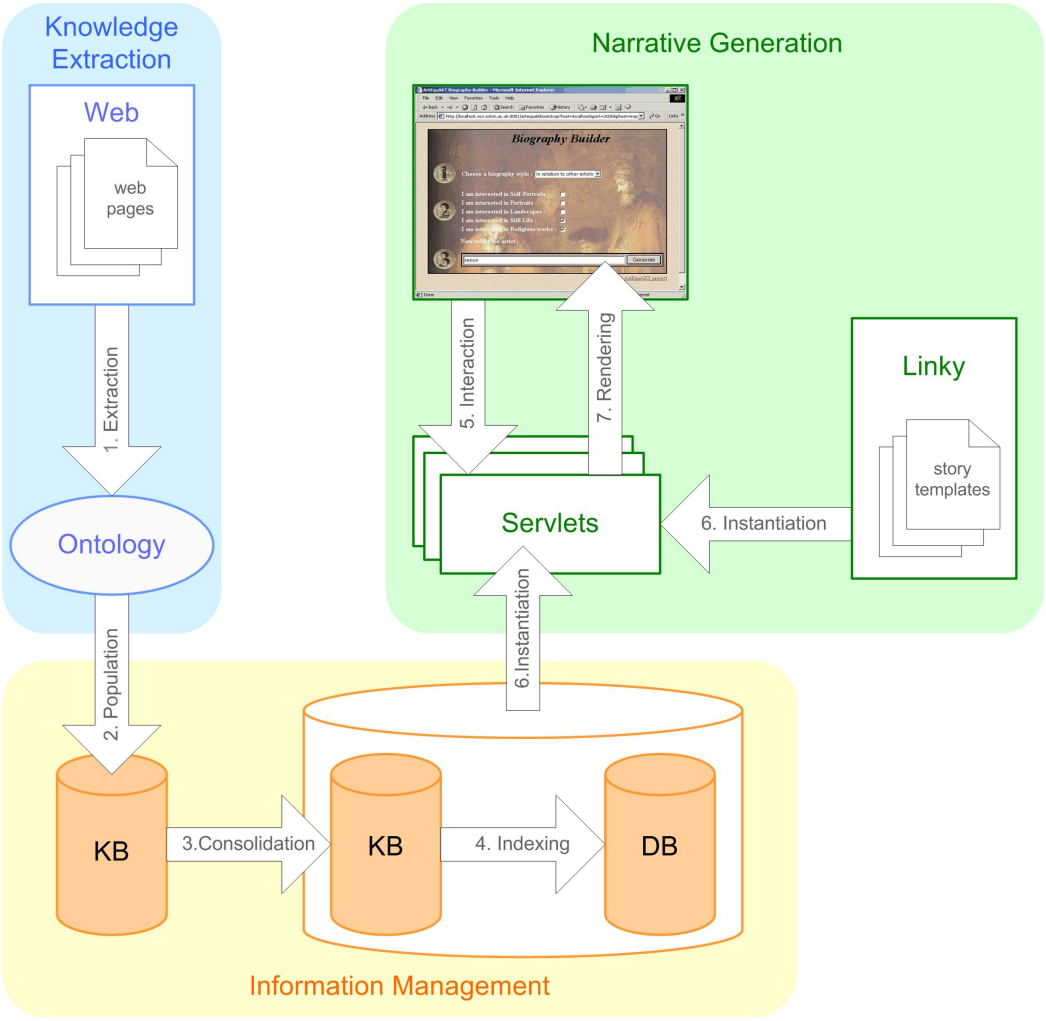


Figure 4: Artequakt's architecture

The first stage of this project consisted of developing an ontology for the domain of artists and paintings. The main part of this ontology was constructed from selected sections in the CIDOC Conceptual Reference Model ontology. The ontology informs the extraction tool of the type of knowledge to search for and extract. An information extraction tool was developed and applied that automatically populates the ontology with information extracts from online documents. The information extraction tool makes use of an ontology, coupled with a general-purpose lexical database, WordNet and an entity-recogniser, GATE (Cunningham et al 2002 – see section 9.4) as guidance tools for identifying knowledge fragments consisting not just of entities, but also the relationships between them. Automatic term expansion is used to increase the scope of text analysis to cover syntactic patterns that imprecisely match our definitions.

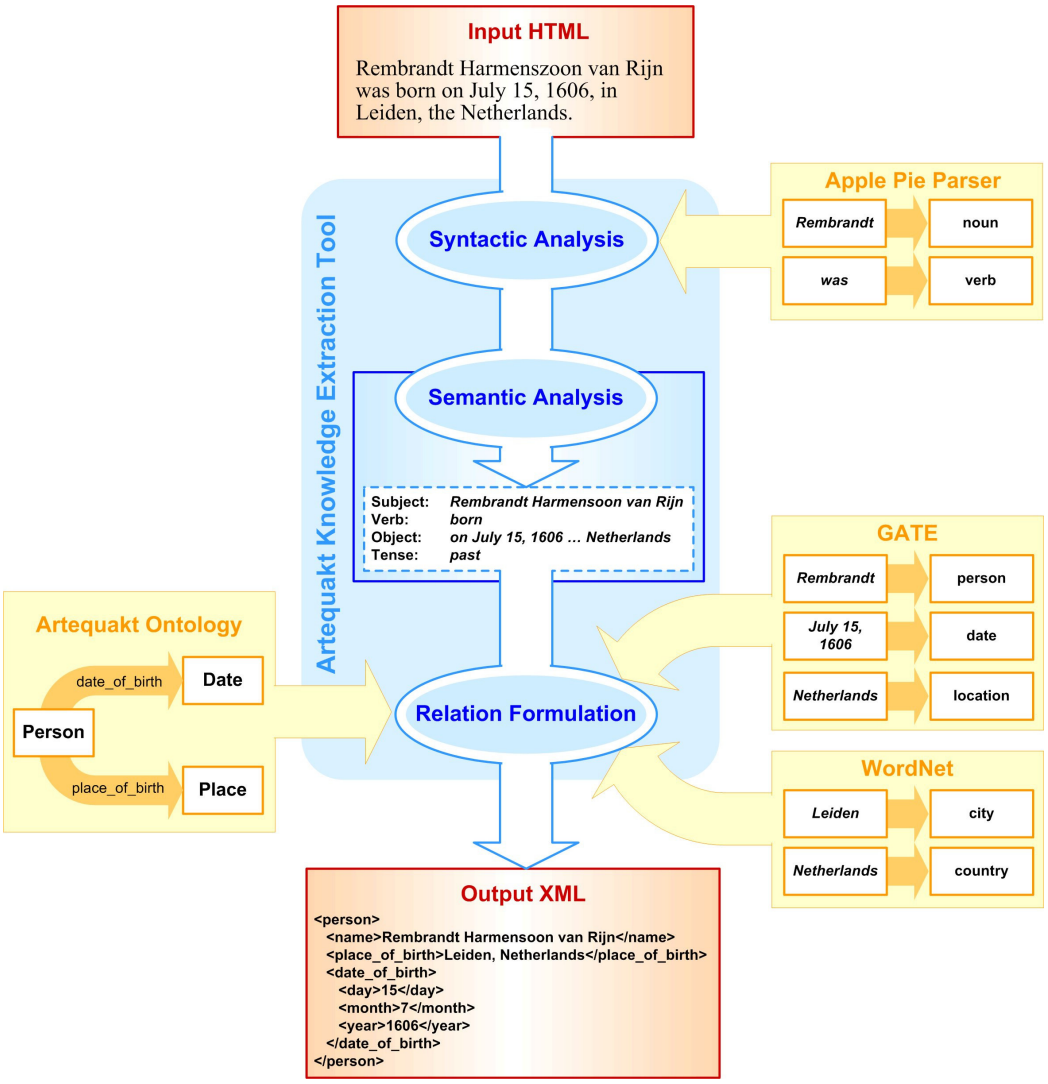


Figure 5: The IE process in Artequakt

The extracted information is stored in a knowledge base and analysed for duplications and inconsistencies. A variety of heuristics and knowledge comparison and term expansion methods were used for this purpose. This included the use of simple geographical relations from WordNet to consolidate any place information; e.g. places of birth or death. Temporal information was also consolidated with respect to precision and consistency.

Narrative construction tools were developed that queried the knowledge base through an ontology server. These queries searched and retrieved relevant facts or textual paragraphs and generated a specific biography. The challenge is to build biographies for artists where there is sparse information available, distributed across the Web. This may mean constructing text from basic factual information gleaned, or combining text from a number of sources with differing interests in the artist. Secondly, the work also aspires to provide biographies that are tailored to the particular interests and requirements of a given reader. These might range from rough stereotyping such as “A biography suitable for a child” to specific reader interests such as “I'm interested in the artists’ use of colour in their oil paintings” (Figure 6).

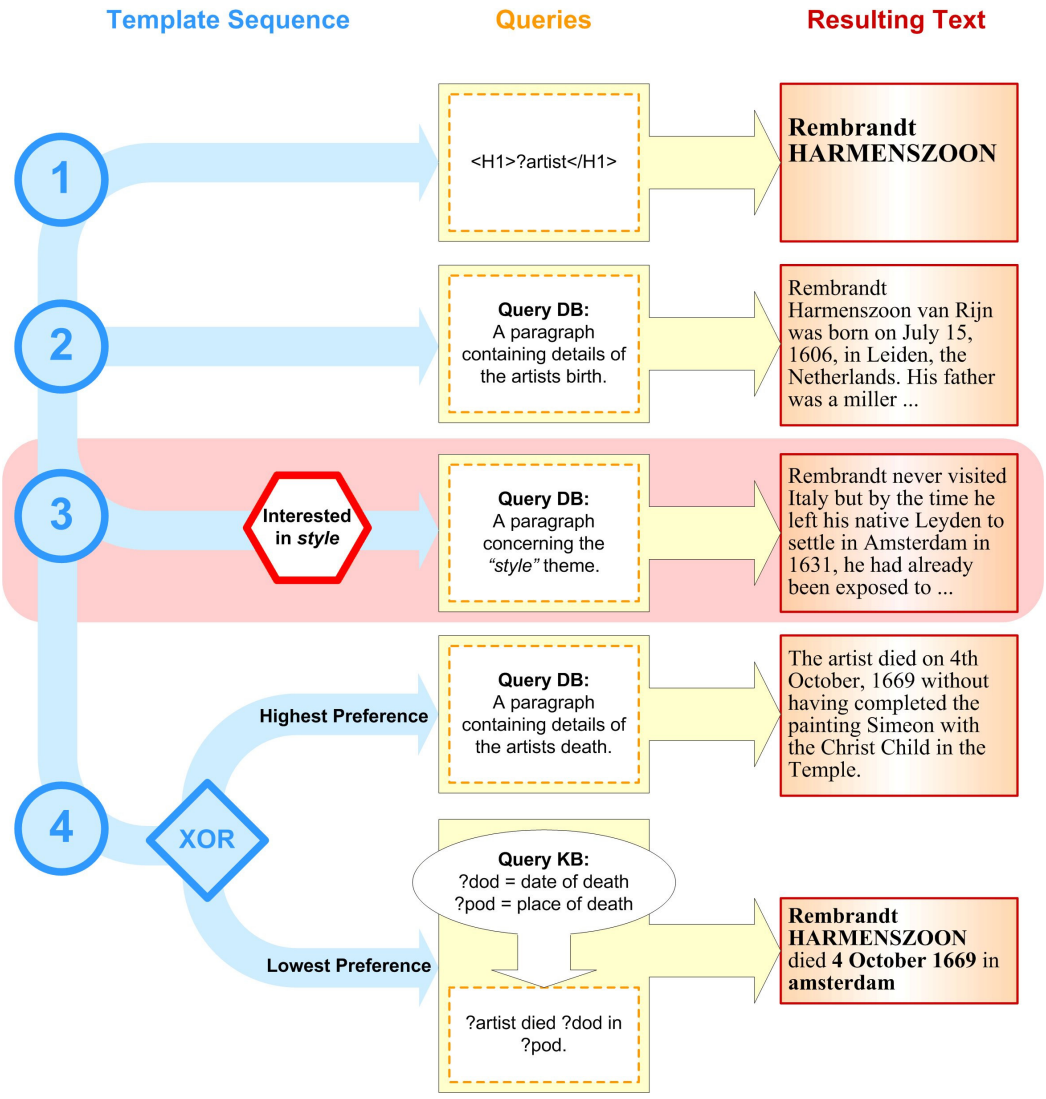
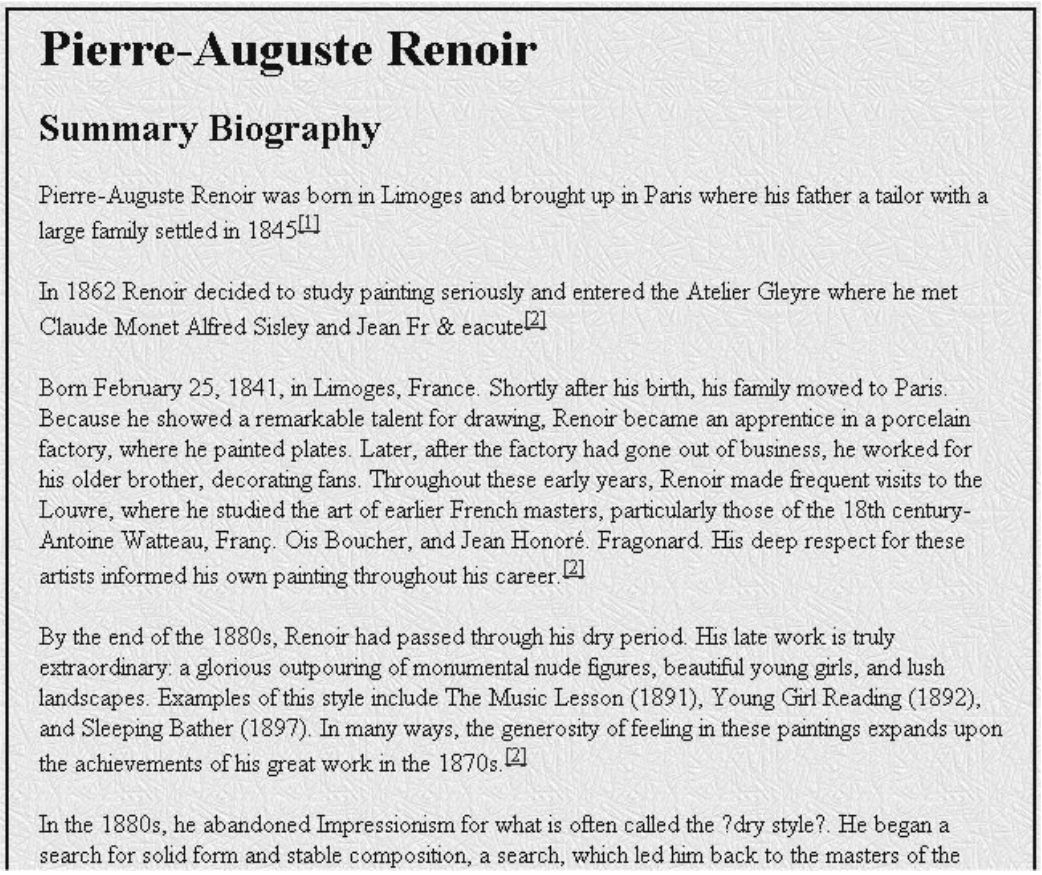


Figure 6: The biography generation process in Artequakt



Pierre-Auguste Renoir

Summary Biography

Pierre-Auguste Renoir was born in Limoges and brought up in Paris where his father a tailor with a large family settled in 1845[1]

In 1862 Renoir decided to study painting seriously and entered the Atelier Gleyre where he met Claude Monet Alfred Sisley and Jean Fr & eacute[2]

Born February 25, 1841, in Limoges, France. Shortly after his birth, his family moved to Paris. Because he showed a remarkable talent for drawing, Renoir became an apprentice in a porcelain factory, where he painted plates. Later, after the factory had gone out of business, he worked for his older brother, decorating fans. Throughout these early years, Renoir made frequent visits to the Louvre, where he studied the art of earlier French masters, particularly those of the 18th century- Antoine Watteau, Franç. Ois Boucher, and Jean Honoré. Fragonard. His deep respect for these artists informed his own painting throughout his career.[2]

By the end of the 1880s, Renoir had passed through his dry period. His late work is truly extraordinary: a glorious outpouring of monumental nude figures, beautiful young girls, and lush landscapes. Examples of this style include The Music Lesson (1891), Young Girl Reading (1892), and Sleeping Bather (1897). In many ways, the generosity of feeling in these paintings expands upon the achievements of his great work in the 1870s.[2]

In the 1880s, he abandoned Impressionism for what is often called the ?dry style?. He began a search for solid form and stable composition, a search, which led him back to the masters of the

Figure 7: Artequakt-generated biography for Renoir

The system is undergoing evaluation and testing at the moment. It has already provided important components for a successful bid (the SCULPTEUR project) into the EU VI Framework.

3.4. Refiner++

Refiner++ (Aiken & Sleeman 2003) is a new implementation of Refiner+ (Winter & Sleeman 1995), an algorithm that detects inconsistencies in a set of examples (cases) and suggests ways in which these inconsistencies might be removed. The domain expert is required to specify which category each case belongs to; Refiner+ then infers a description for each of the categories and reports any inconsistencies that exist in the dataset. An inconsistency is when a case matches a category other than the one in which the expert has classified it. If inconsistencies have been detected in the dataset, the algorithm attempts to suggest appropriate ways of dealing with the inconsistencies by refining the dataset. At the time of writing, the Refiner++ system has been presented to three experts to use on problems in their domains: anaesthetics, educational psychology, and intensive care.

Although the application can be used to import existing datasets and perform analysis on them, its real strength is for an expert who wants to conceptualize a domain where the inherent task is classification. Refiner++ requires the expert to articulate cases, specifying the descriptors they believe to be important in their domain. This causes the expert to conceptualize their domain, bringing out the hidden relationships between descriptors that might otherwise be ignored.

We hope to produce a "refinement workbench" to include Refiner++, ReTax (Alberdi & Sleeman 1997) and ConRef (Winter et al 1998 – and section 8.2).

4. Modelling

As noted in the previous section, ontologies loom large in AKT – as in the SW – for modelling purposes. In particular, we have already seen the importance of ontologies (a) for directing acquisition, and (b) as objects to be acquired or created. The SW, as we have argued, will be a domain in which services will be key. For particular tasks, agents are likely to require combinations of services, either in parallel or sequentially. In either event, ontologies are going to be essential to provide shared understandings of the domain, preconditions and postconditions for their application and optimal combination. However, in the likely absence of much standardisation, ontologies are not going to be completely shared.

Furthermore, it will not be possible to assume unique or optimal solutions to the problem of describing real-world contexts. Ontologies will be aimed at different tasks, or will make inconsistent, but reasonable, assumptions. Given two ontologies precisely describing a real-world domain, it will not in general be possible to guarantee mappings between them without creating new concepts to augment them. As argued in (Kalfoglou & Schorlemmer 2003a), and section 8.3 above, there is a distinct lack of formal underpinnings here. Ontology mapping will be an important aspect to knowledge modelling, and as we have already seen, AKT is examining these issues closely.

Similarly, the production of ontologies will need to be automated, and documents will become a vital source for ontological information. Hence tools such as Adaptiva (section 3.2) will, as we have argued, be essential. However, experimental evidence amassed during AKT shows that in texts, it is often the essential ontological information that is not *expressed*, since it is taken to be part of the 'background knowledge' implicitly shared by reader and author (Brewster et al 2003). Hence the problem of how to augment information sources for documents is being addressed by AKT (Brewster et al 2003).

A third issue is that of the detection of errors in automatically-extracted ontologies, particularly from unstructured material. It was for these reasons that we have also made some attempts to extract information from semi-structured sources ie programs and Knowledge Bases (Sleeman et al, 2003).

In all these ways, there are plenty of unresolved research issues with respect to ontologies that AKT will address over the remaining half of its remit. However, modelling is a fundamental requirement in other areas, for instance with respect to the modelling of business processes in order to achieve an understanding of the events that a business must respond to. The AKT consortium has amassed a great deal of experience of modelling processes such as these that describe the context in which organisations operate. Section 4.1 looks at the use of protocols to model service interactions, while in section 4.2 we will briefly discuss one use of formal methods to describe lifecycles.

4.1. Service interaction protocol

In an open system, such the Semantic Web, communication among agents will, in general, be asynchronous in nature: the imposition of synchronicity would constrain agent behaviour and require additional (and centralised) infrastructure. However,

asynchronous communication can fail in numerous ways – messages arrive out of sequence, or not at all, agents fail in undetermined states, multiple dialogues are confused, perhaps causing agents to adopt mistaken roles in their interactions, thereby propagating the failure through its future communications. The insidious nature of such failures is confirmed by the fact that their causes – and sometimes the failures themselves – are often undetectable.

To address this problem, the notion of *service interaction protocols* has been developed. These are formal structures representing distributed dialogues which serve to impose social conventions or *norms* on agent interactions (cf the *communications policy* work of Bradshaw and his colleagues at IHMC). A protocol specifies all possible sequences of messages that would allow the dialogue to be conducted successfully within the boundaries established by the norm. All agents engaging in the interaction are responsible for maintaining this dialogue, and the updated dialogue is passed in its entirety with each communication between agents. Placing messages in the context of the particular norm to which they relate in this manner allows the agents to understand the current state of the interaction and locate their next roles within it, and so makes the interactions in the environment more resistant to the problems of asynchrony.

Furthermore, since these protocols are specified in a formal manner, they can be subjected to formal model checking as well as empirical (possibly synchronous) 'off-line' testing before deployment. In addition to proving certain properties of dialogues are as desired, this encourages the exploration of alternative descriptions of norms and the implications these would have for agent interactions (Vasconcelos et.al. 2002, Walton & Robertson 2002).

In addition to this work on service interaction protocols, we have also encountered the issues of service choreography in an investigation of the interactions between the Semantic Web, the agent-based computing paradigm and the Web Services environment.

The predominant communications abstraction in the agent environment is that of speech acts or performatives, in which inter-agent messages are characterised according to their perlocutionary force (the effect upon the listener). Although the Web Services environment does not place the same restrictions on service providers as are present on agents (in which an agent's state is modelled in terms of its beliefs, desires and intentions, for example), the notion of performative-based messages allows us abstract the effects, and expectations of effect, of communications.

The set of speech acts which comprise an agent's communicative capabilities in an agent-based systems is known as an agent communication language. In (Gibbins et al 2003), we describe the adaptation of the DAML Services ontology for Web Service description to include an agent communication language component. This benefits service description and discovery by separating the application domain-specific contents of messages from their domain-neutral pragmatics, and so simplifying the design of brokerage components which match service providers to service consumers. This work was carried out in collaboration with QinetiQ, and was realised in a prototype system for situational awareness in a simulated humanitarian aid scenario.

Formal models provide an interesting method for understanding business processes. In the next section, we look at their use to describe knowledge system lifecycles.

4.2. A lifecycle calculus

Knowledge-intensive systems have lifecycles. They are created through processes of knowledge acquisition and problem solver design and reuse; they are maintained and adapted; and eventually they are decommissioned. In software engineering, as well as in the more traditional engineering disciplines, the study of such processes and their controlled integration through the lifetime of a product is considered essential and provides the basis for routine project management activities, such as cost estimation and quality management. As yet, however, we have not seen the same attention to life cycles in knowledge engineering.

Our current need to represent and reason formally about knowledge lifecycles is spurred by the Internet, which is changing our view of knowledge engineering. In the past we built and deployed reasoning systems which typically were self-contained, running on a single computer. Although these systems did have life cycles of design and maintenance, it was only necessary for these to be understood by the small team of engineers who actually were supporting each system. This sort of understanding is close to traditional software engineering so there was no need to look beyond traditional change management tools to support design. Formal representation and reasoning was confined to the artefacts being constructed, not to the process of constructing them. This situation has changed. Ontologies, knowledge bases and problem solvers are being made available globally for use (and reuse) by anyone with the understanding to do so. But this raises the problem of how to gain that understanding. Even finding appropriate knowledge components is a challenge. Assembling them without any knowledge of their design history is demanding. Maintaining large assemblies of interacting components (where the interactions may change the components themselves) is impossible in the absence of any explicit representation of how they have interacted.

4.2.1 The value of formality

There is, therefore, a need for formality in order to be able to provide automated support during various stages of the knowledge-management life cycle. The aim of formality in this area is twofold: to give a concise account of what is going on, and to use this account for practical purposes in maintaining and analysing knowledge-management life cycles. If, as envisioned by the architects of the Semantic Web, knowledge components are to be made available on the Internet for humans and machines to use and reuse, then it is natural to study and record the sequences of transformations performed upon knowledge components. Agents with the ability to understand these sequences would be able to know the provenance of a body of knowledge and act accordingly, for instance, by deciding their actions depending on their degree of trust in the original source of a body of knowledge, or of the specific knowledge transformations performed on it.

Different sorts of knowledge transformations preserve different properties of the components to which they are applied. Being able to infer such property-preservation from the structure of a life cycle of a knowledge component may be useful for agents, as it can help them decide which reasoning services to use in order to perform deductions without requiring the inspection of the information contained in knowledge components themselves.

Knowing whether these kinds of properties are preserved across life cycles would be useful, especially in environments such as the WWW, where knowledge components

are most likely to be translated between different languages, mapped into different ontologies, and further specialised or generalised in order to be reused together in association with other problem solvers or in other domains. Thus, having a formal framework with a precise semantics in which we could record knowledge transformations and their effect on certain key properties would allow for the analysis and automation of services that make use of the additional information contained in life-cycle histories.

4.2.2 The AKT approach

We have been exploring a formal approach to the understanding of lifecycles in knowledge engineering. Unlike many of the informal life-cycle models in software engineering, our approach allows for a high level of automation in the use of lifecycles. When supplied with appropriate task-specific components, it can be deployed to fully automate life-cycle processes. Alternatively, it can be used to support manual processes such as reconstruction of chains of system adaptation. We have developed a formal framework for describing life cycles and mechanisms for using these by means of a *lifecycle calculus* of abstract knowledge transformations with which to express life cycles of knowledge components.

To allow us to operate at an abstract level, without committing ourselves to a particular knowledge representation formalism or a particular logical system, we have based our treatment of knowledge transformation on abstract model theory. In particular, we use *institutions* and *institution morphisms* (Goguen & Burstall 1992) as mathematical tools upon which to base a semantics of knowledge components and their transformations. An institution captures the essential aspects of logical systems underlying any specification theory and technology. In practice, the idea of a single one-size-fits-all life-cycle model is implausible. Applications of formal knowledge life cycles may use more specialised calculi and use these to supply different degrees of automated support.

In (Schorlemmer et al 2002a) we show how to reason about properties that are or are not preserved during the life cycle of a knowledge component. Such information may be useful for the purposes of high-level knowledge management. If knowledge services that publicise their capabilities in distributed environments, such as the Web, also define and publicise the knowledge transformations they realise in terms of a formal language, then automatic brokering systems may use this additional information in order to choose among several services according to the properties one would like to preserve.

In (Schorlemmer et al 2002b) we analyse a real knowledge-engineering scenario consisting of the life cycle of an ontology for ecological meta-data, and describe it it terms of our life-cycle calculus. We show how this could be done easily with the support of a life-cycle editing tool, F-Life (Robertson & Schorlemmer 2003), that constructs formal life-cycle patterns by composing various life cycle rules into a set of Horn clauses that constitute a logic program. This program can then be used to recreate with a meta-interpreter analogous life-cycles, following the same steps we have previously compiled by means of the editor. We also describe an architecture in which the brokering of several knowledge services in a distributed environment is empowered by the additional information we obtain from formal life-cycle patterns. In particular we show how the previously edited abstract life-cycle pattern can be used to guide a brokering system in the task of choosing the appropriate problem solvers in order to execute a concrete sequence of life-cycle steps. The information of the

concrete life cycle that is followed is then stored alongside the transformed knowledge component, so that this information may subsequently be used by other knowledge services.

5. Reuse

Reuse is, of course, a hallowed principle of software engineering, that has certainly been adopted in knowledge management. And, of course, given the problems of knowledge acquisition– the KA bottleneck – and the difficulties with, for example, the creation and maintenance of ontologies that we have already noted in earlier sections, it clearly makes sense to reuse expensively-acquired knowledge or models etc in KM.

However, as always, such things are easier said than done. Many KM artefacts are laboriously handcrafted, and as such require a lot of rejigging for new contexts. Automatically-generated material also carries its own problems. Selection of material to reuse is also a serious issue. But with all the knowledge lying around, say, on the WWW, the power of the resource is surely too great to be ignored. Hence reuse is a major knowledge challenge in its own right, which AKT has been investigating. As one example, AKT has been investigating the reuse of expensively-acquired constraints, and also of various pre-existing knowledge services, in the management of a virtual organization (section 5.1). Furthermore, if services and/or resources are to be reused, then the user will technically have a large number of possible services for any query – if queries are composed, the space of possible combinations could become very large. Hence, brokering services will be of great importance, and AKT's investigations of this concept are reported in section 5.2. Work will also have to be done on the modification and combination of knowledge bases, and we will see tools for this in section 5.3.

5.1. KRAFT/I-X

Virtual organizations are the enabling enterprise structure in modern e-business, e-science, and e-governance. Such organizations most effectively harness the capabilities of individuals working in different places, with different expertise and responsibilities. Through the communication and computing infrastructure of the virtual organization, these people are able to work collaboratively to accomplish tasks, and together to achieve common organisational goals. In the KRAFT/I-X Technology Integration Experiment (TIE), a number of knowledge-based technologies are integrated to support workers in a virtual organization (cf Figure 8):

- *Workflow* and *business process modelling* techniques (Chen-Burger and Stader 2003, Chen-Burger et al 2002) provide the coordination framework to facilitate smooth, effective collaboration among users;
- *Task-supporting user interfaces* (I-X process panels – Tate 2003, and http://www.aktors.org/technologies/ix/).
- *Constraint interchange and solving* techniques (Gray et al 2001, Hui et al 2003) guide users towards possible solutions to shared problems, and keep the overall state of the work activity consistent;
- *Agent-based infrastructure* provides the underlying distributed, heterogeneous software architecture (the AKTbus – http://www.aktors.org/technologies/aktbus/).

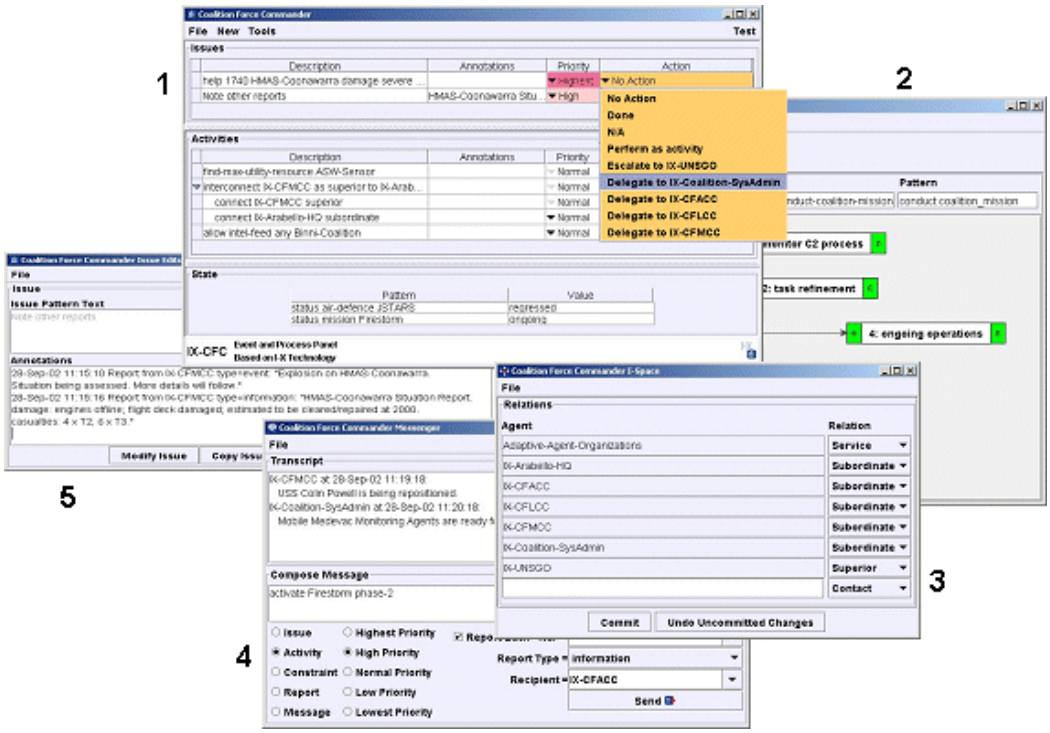


Figure 8: The I-X Tools include: 1. Process Panel (I-P2); 2. Domain Editor (I-DE): create and modify process models; 3 I-Space: maintain relationships with other agents; 4 Messenger: instant messaging tool, for both structured and less formal communications; 5 Issue Editor: create, modify, annotate issues.

As a simple example of an application that the KRAFT/I-X TIE can support, consider a Personal Computer purchasing process in an organization. There will typically be several people involved, including the end-user who needs a PC, a technical support person who knows what specifications and configurations are possible and appropriate, and a financial officer who must ensure that the PC is within budget. In the implemented KRAFT/I-X demonstrator, the second and third of these people are explicitly represented: the technical support by a process panel running in Aberdeen (ABDN-panel, Figure 9) and the finance officer by a panel running in Edinburgh (ED-panel). Note that the user is represented implicitly by the PC requirements input to the system through the ED-panel. The two panels share a workflow/business process model that enables them to cooperate. As part of this workflow, the ED-panel passes user requirement constraints to the ABDN-panel, so that a feasible technical configuration for the PC can be identified. In fact, the ABDN-panel uses a knowledge base of PC configurations, and a constraint-solving system (KRAFT) to identify the feasible technical configuration, which is then passed back to the ED-panel via the ABDN-panel.

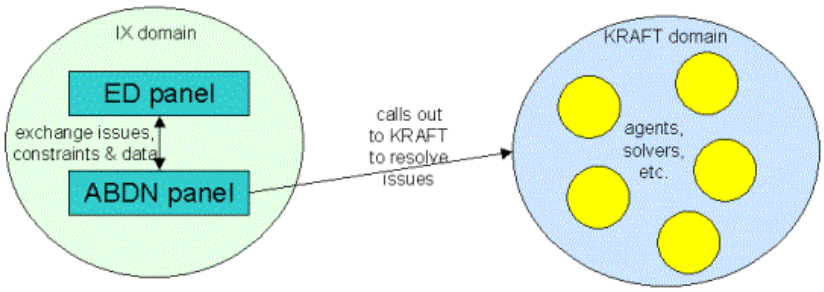


Figure 9: KRAFT-I/X demonstrator architecture

The various components of the KRAFT/I-X implementation communicate by means of a common knowledge-interchange protocol (over AKTbus) and an RDF-based data and constraint interchange format (Hui et al 2003). The AKTbus provides a lightweight XML-based messaging infrastructure and was used to integrate a number of pre-existing systems and components from the consortium members, as described more fully in section 9.1.

5.2. The AKT broker

In order to match service requests with appropriate Semantic Web services (and possibly sequences of those services), some sort of brokering mechanism would seem to be needed. Service-providing agents advertise to this broker a formal specification of each offered service in terms of its inputs, outputs, preconditions, effects, and so on. This specification is constructed using elements from one or more shared ontologies, and is stored within the broker. When posted to the broker, a request – in the form of the specification of the desired service – is compared to the available services for potential matches (and it may be possible – and sometimes necessary – to compose sequences of several services to meet certain requests).

However, this approach to service brokering raises a number of practical questions. As for all techniques dependent on shared ontologies, the source and use of these ontologies is an issue. And with brokering there is a particular problem concerning the appropriate content of service specifications: rich domain ontologies make possible rich specifications – and also increase the possible search space of services and the reasoning effort required to determine if service and request specification match. One solution to this, it might be thought, is to constrain the ontologies to describe very specific service areas, thereby constraining the specification language. Some focusing of ontologies in this manner may be desirable, resulting in a broker that is specialised for particular services or domains rather than being general-purpose. However, if the constraints placed on ontologies are too great this will result in very specialised brokers, and would have the effect of shifting the brokering problem from one of finding appropriate services to one of finding appropriate service *brokers* – and so, some sort of 'meta-brokering' mechanism would be necessary, and the brokering problem would have to be addressed all over again.

While careful ontological engineering would appear unavoidable, alternative approaches to this problem that we have been investigating involve using ideas emerging elsewhere in the project to prune the search space. For example, by encouraging the description of services in terms of the lifecycle calculus (section 5.2), where appropriate, to complement their specifications, allows additional constraints to be placed on service requests and the search for matching services to be focused upon those conforming to these constraints. Likewise, considering the brokering task as

being, in effect, one of producing an appropriate service interaction protocol for the request, can serve to concentrate the search on to those services that are willing and able to engage in such protocols.

5.3. Reusing knowledge bases

Finally, the facilitation of reuse demands tools for identifying, modifying and combining knowledge bases for particular problems. In this section, we look at MUSKRAT and ConcepTool for addressing these issues.

5.3.1 MUSKRAT

MUSKRAT (Multistrategy Knowledge Refinement and Acquisition Toolbox – White & Sleeman 2000) aims to unify problem solving, knowledge acquisition and knowledge-base refinement in a single computational framework. Given a set of Knowledge Bases (KBs) and Problem Solvers (PSs), the MUSKRAT-Advisor investigates whether the available KBs will fulfil the requirements of the selected PS for a given problem. We would like to reject impossible combinations KBs and PSs quickly. We represent combinations of KBs and PSs as CSPs. If a CSP is not consistent, then the combination does not fulfil the requirements. The problem then becomes one of quickly identifying inconsistent CSPs. To do this, we propose to relax the CSPs: if we can prove that the relaxed version is inconsistent then we know that the original CSP is also inconsistent. It is not obvious that solving relaxed CSPs is any easier. In fact, phase transition research (e.g. Prosser 1994) seems to indicate the opposite when the original CSP is inconsistent. We have experimented with randomly generated CSPs (Nordlander et al 2002), where the tightness of the constraints in a problem varies uniformly. We have shown that careful selection of the constraints to relax can save up to 70% of the search time. We have also investigated practical heuristics for relaxing CSPs. Experiments show that the simple strategy of removing constraints of low tightness is effective, allowing us to save up to 30% of the time on inconsistent problems without introducing new solutions.

In the constraints area, future work will look at extending this approach to more realistic CSPs. The focus will be on scheduling problems, which are likely to involve non-binary and global constraints, and constraint graphs with particular properties (e.g. Walsh 2001). We will also investigate more theoretical CSP concepts, including higher consistency levels and problem hardness. Success in this research will allow us to apply constraint satisfaction and relaxation techniques to the problem of knowledge base reuse.

5.3.2 ConcepTool

ConcepTool is an Intelligent Knowledge Management Environment for building, modifying, and combining expressive domain knowledge bases and application ontologies. Apart from its user-oriented editing capabilities, one of the most notable features of the system is its extensive automated support to the analysis of knowledge being built, modified or combined. ConcepTool uses Description Logic-based taxonomic reasoning to provide analysis functionalities such as KB consistency, detection of contradicting concepts, making explicit of hidden knowledge and ontology articulation.

The development of the core ConcepTool system has been funded on a separate grant by the EPSRC, while the development of the articulation functionalities has been funded by the AKT IRC consortium. Notably, two systems have been actually

developed: the first one, which supported modelling and analysis on an expressive Enhanced Entity-Relationship knowledge model, has been used as a prototype for the development of the second one, which uses a frame-based model. Both versions of ConcepTool can handle complex, sizeable ontologies (such as the AKT one), supporting the combination of heterogeneous knowledge sources by way of taxonomic, lexical and heuristic analysis.

6. Retrieval

Given the amount of information available on the WWW, clearly a major problem is retrieving that information from the noise which surrounds it. Retrieval from large repositories is a major preoccupation for AKT. There is a major trend, supported by the Semantic Web, towards annotating documents, which should enable more intelligent retrieval (section 6.2). Furthermore, such annotations will facilitate the difficult problem, already apparent under several of our headings above, of ontology population.

However, annotation itself will not solve all the problems of information retrieval. Information is often dispersed, or distributed, around large unstructured repositories – like the WWW itself – in such a way as to make systematic retrieval impossible, and intelligent retrieval difficult. Information may indeed only be implicit in repositories, in which case retrieval must include not only the ability to locate the important material, but also the ability to perform inference on it (while avoiding circularity – how does one identify the important information prior to inferring about the representation that contains it in implicit form?). As well as unstructured, distributed repositories, information can also be hidden in unstructured *formats*, such as plain text or images (section 6.1).

However, even information held in relatively structured formats can be hard to get at, often because it is implicit. One issue that AKT has been addressing here is that of extracting information from ontologies about structures within organisations, in particular trying to extract implicit information about informal communities of interest or practice based on more formal information about alliances, co-working practices, etc (section 6.3).

6.1. Ontology-based information extraction

6.1.1 Amilcare

Information extraction from text (IE) is the process of populating a structured information source (e.g. an ontology) from a semi-structured, unstructured, or free text, information source. Historically, IE has been seen as the process of extracting information from newspaper-like texts to fill a template, i.e. a form describing the information to be extracted.

We have worked in the direction of extending the coverage of IE first of all to different types of textual documents, from rigidly structured web pages (e.g. as generated by a database) to completely free (newspaper-like) texts, with their intermediate types and mixtures (Ciravegna 2001a).

Secondly we have worked on the use of machine learning for allowing porting to different applications using domain specific (non linguistic) annotation. The result is the definition of an algorithm—called $(LP)^2$ (Ciravegna 2001b) and (Ciravegna

2001c) — able to cope with a number of types of IE tasks on different types of documents using only domain-specific annotation.

Amilcare (Ciravegna and Wilks 2003) is a system that has been defined using $(LP)^2$ that is specifically designed for IE for document annotation. Amilcare has become the basis of assisted annotation for the Semantic Web in three tools: Melita, MnM (both developed as part of AKT – see section 6.2.1) and Ontomat (Handschuh et al. 2002).

Amilcare has been also released to some 25 external users, including a dozen companies, for research. It is also, as can be seen in references throughout this paper, central to many AKT technologies and services.

6.1.2 AQUA

AQUA (Vargas-Vera et al in press) is an experimental question answering system. AQUA combines Natural Language processing (NLP), Ontologies, Logic, and Information Retrieval technologies in a uniform framework. AQUA makes intensive use of an ontology in several parts of the question answering system. The ontology is used in the refinement of the initial query, the reasoning process (a generalization/specialization process using classes and subclasses from the ontology), and in the novel similarity algorithm. The similarity algorithm, is a key feature of AQUA. It is used to find similarities between relations/concepts in the translated query and relations/concepts in the ontological structures. The similarities detected then allow the interchange of concepts or relations in a logic formula corresponding to the user query.

6.2. Annotation

Amilcare and $(LP)^2$ constitute the basis upon which the AKT activity on IE has been defined. It mainly concerns annotation for the SW and KM. The SW needs semantically-based document annotation to both enable better document retrieval and empower semantically-aware agents. Most of the current technology is based on human centered annotation, very often completely manual (Handschuh et al 2002). Manual annotation is difficult, time consuming and expensive (Ciravegna et al 2002).

Convincing millions of users to annotate documents for the Semantic Web is difficult and requires a world-wide action of uncertain outcome. In this framework, annotation is meant mainly to be statically associated to (and saved within) the documents. Static annotation associated to a document can:

(1) be incomplete or incorrect when the creator is not skilled enough;

(2) become obsolete, i.e. not be aligned with pages' updates;

(3) be irrelevant for some use(r)s: a page in a pet shop web site can be annotated with shop-related annotations, but some users would rather prefer to find annotations related to animals.

Producing methodologies for automatic annotation of pages therefore becomes important: the initial annotation associated to the document loses its importance because at any time it is possible to automatically (re)annotate the document. Also documents do not need to contain the annotation, because it can be stored in a separate database or ontology exactly as nowadays' search engines do not modify the indexed documents. In the future Semantic Web, automatic annotation systems might become as important as indexing systems are nowadays for search engines.

Two strands of research have been pursued for annotation: assisted semi-automatic document annotation (mainly suitable for knowledge management) and unsupervised annotation of large repositories (mainly suitable for the Semantic Web).

6.2.1 Assisted annotation

AKT has developed assisted annotation tools that can be used to create an annotation engine. They all share the same method based on adaptive IE (Amilcare). In this sections, we will describe two tools: MnM (Vargas-Vera et al. 2002) and Melita (Ciravegna et al. 2002) – though see also the sections on Magpie (section 6.2.2) and CS AKTive Space(section 10.1).

In both cases annotation is ontology-based. The annotation tool is used to annotate documents on which the IE system trains. The IE system monitors the user-defined annotations and learns how to reproduce it by generalizing over the seen examples. Generalization is obtained by exploiting both linguistic and semantic information from the ontology.

MnM focuses more on the aspect of ontology population. Melita has a greater focus on the annotation lifecycle.

MnM

The MnM tool supports automatic, semi-automatic and manual semantic annotation of web pages. MnM allows users to select ontologies, either by connecting to an ontology server or simply through selection of the appropriate file, and then allows them to annotate a web resource by populating classes in the chosen ontology with domain specific information.

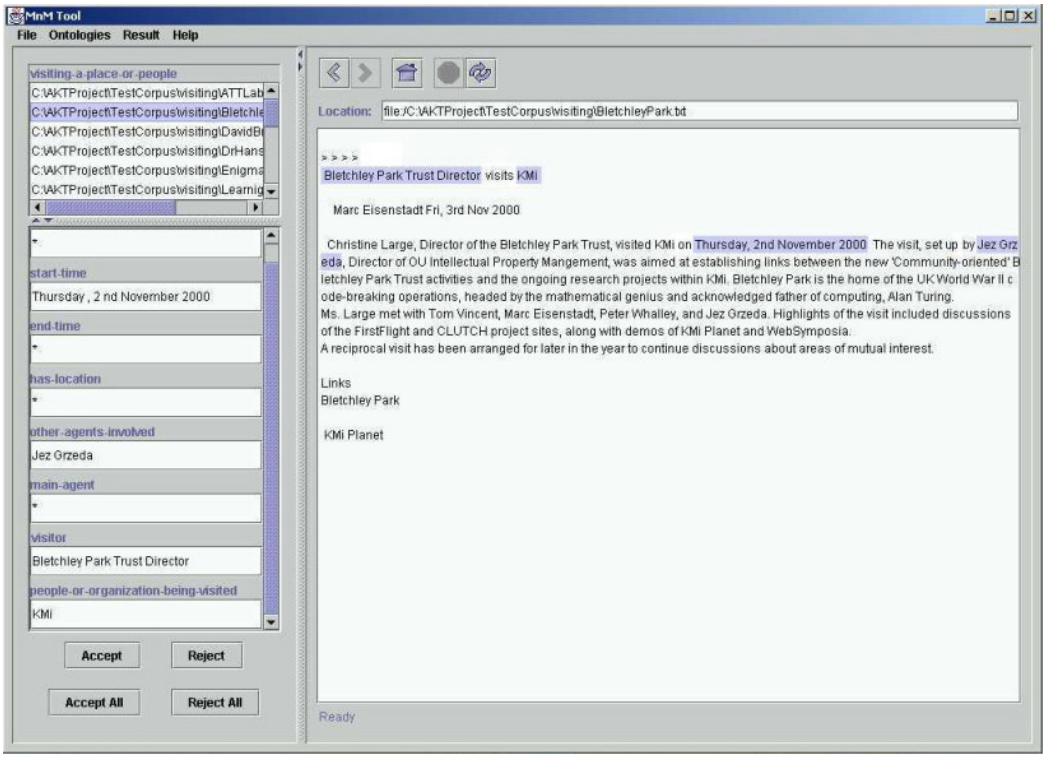


Figure 10 A Screenshot of the MnM Annotation Tool

An important aspect of MnM is the integration with information extraction technology to support automated and semi-automated annotation. This is particularly important as manual annotation is only feasible in specifc contexts, such as high-value e-commerce applications and intranets. Automated annotation is achieved through a generic plug-in mechanism, which is independent of any particular IE tool, and which has been tested with Amilcare. The only knowledge required for using Amilcare in new domains is the ability of manually annotating the information to be extracted in a training corpus. No knowledge of Human Language technologies is necessary.

MnM supports a number of representation languages, including RDF(S), DAML+OIL and OCML. An OWL export mechanism will be developed in the near future. MnM has been released open source and can be downloaded from http://kmi.open.ac.uk/projects/akt/MnM/. This version of MnM also includes a customized version of Amilcare.

Melita

Melita is a tool for defining ontology-based annotation tools. It uses Amilcare as active support to annotation. The annotation process is based on a cycle that includes:

(1) The manual definition or revision of a draft ontology;

(2) The (assisted) annotation of a set of documents; initially the annotation is completely manual, but Amilcare runs in the background and learns how to annotate. Once Amilcare has started to learn, it preannotates every new text before Melita presents it to the user; the user must correct the system annotation; corrections (missed and wrong cases) are sent back to Amilcare for retraining.

(3) Go to 1., until the IE system has reached a sufficient reliability in the annotation process and the annotation service is delivered.

In this process, users may eventually decide to try to write annotation rules themselves either to speed up the annotation process or to help the IE system learning (e.g. by modifying the induced grammar).

Melita provides three centers of focus of user interaction for supporting this lifecycle:

- the ontology;
- the corpus, both as a whole and as a collection of single documents;
- the annotation pattern grammar(s), either user- or system-defined.

Users can move the focus and the methodology of interaction during the creation of the annotation tool in a seamless way, for example moving from a focus on document annotation (to support rule induction or to model the ontology), to rule writing, to ontology editing (Ciravegna et al. 2003 submitted).

6.2.2 Annotation of large repositories

Armadillo

The technology above can only be applied when the documents to be analyzed present some regularity in terms of text types and recurrent patterns of information. This is sometimes but not always the case when we look at companies' repositories. In the event that texts are very different or highly variable in nature (e.g. on the Web), the

Melita approach is inapplicable, because it would require the annotation of very large corpora, a task mostly unfeasible.

For this reason, AKT has developed a methodology able to learn how to annotate semantically consistent portions of the Web in a complete unsupervised way, extracting and integrating information from different sources. All the annotation is produced automatically with no user intervention apart from some corrections the users might want to perform to the system's final or intermediate results. The methodology has been fully implemented in Armadillo, a system for unsupervised information extraction and integration from large collections of documents (http://www.aktors.org/technologies/Armadillo/) (Ciravegna et al. 2003).

The natural application of such methodology is the Web, but very large companies' information systems are also an option.

The key feature of the Web exploited by the methodology is the *redundancy* of information. Redundancy is given by the presence of multiple citations of the same information in different contexts and in different superficial formats, e.g., in textual documents, in repositories (e.g. databases or digital libraries), via agents able to integrate different information sources, etc. From them or their output, it is possible to extract information with different reliability. Systems such as databases generally contain structured data and can be queried using an API. In case the API is not available (e.g. the database has a Web front end and the output is textual), wrappers can be induced to extract such information (Kushmerick et al. 1997). When the information is contained in textual documents, extracting information requires more sophisticated methodologies. There is an obvious increasing degree of complexity in the extraction task mentioned above. The more difficult the task, the less reliable generally the extracted information is. For example wrapper induction systems generally reach 100% on rigidly structured documents, while IE systems reach some 70% on free texts. Also, the more the complexity increases, the more the amount of data needed for training grows: wrappers can be trained with a handful of examples whereas full IE systems may require millions of words.

In our model, learning of complex modules is bootstrapped by using information from simple reliable sources of information. This information is then used to annotate documents to train more complex modules. A detailed description of the methodology can be found in (Ciravegna et al. 2003).

Magpie

Automatic annotation could also be the key to improving strategies and information for browsing the SW. This is the intuition behind Magpie (Dzbor et al 2003). Web browsing involves two basic tasks: (i) finding the right web page and (ii) *making sense* of its content. A lot of research has gone into supporting the task of finding web resources, either by means of 'standard' information retrieval mechanisms, or by means of semantically enhanced search (Gruber 1993, Lieberman et al 2001). Less attention has been paid to the second task – supporting the *interpretation* of web pages. Annotation technologies allow users to associate meta-information with web resources, which can then be used to facilitate their interpretation. While such technologies provide a useful way to support group-based and shared interpretation, they are nonetheless very limited; mainly because the annotation is carried out manually. In other words, the quality of the sensemaking support depends on the willingness of stakeholders to provide annotation, and their ability to provide valuable

information. This is of course even more of a problem, if a formal approach to annotation is assumed, based on semantic web technology.

Magpie follows a different approach from that used by the aforementioned annotation techniques: it automatically associates a semantic layer to a web resource, rather than relying on a manual annotation. This process relies on the availability of an ontology. Magpie offers complementary knowledge sources, which a reader can call upon to quickly gain access to any background knowledge relevant to a web resource. Magpie may be seen as a step towards a *semantic web browser*.

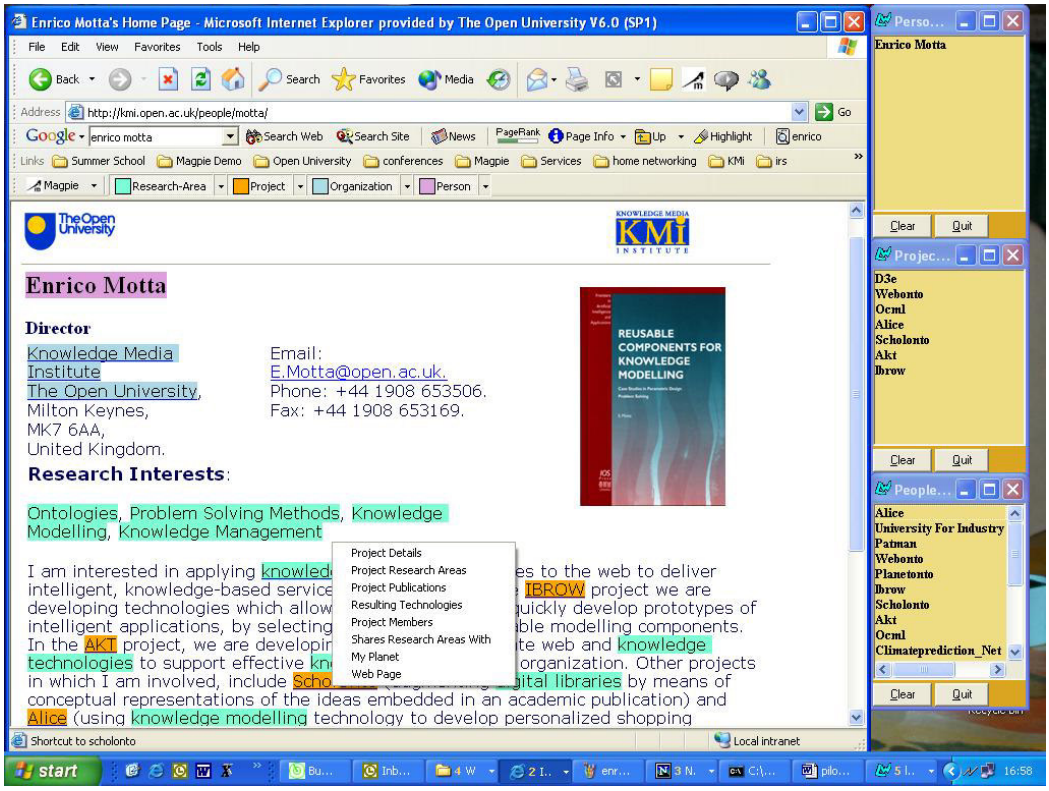


Figure 11 The Magpie Semantic Web Browser

The Magpie-mediated association between an ontology and a web resource provides an interpretative viewpoint or context over the resource in question. Indeed the overwhelming majority of web pages are created within a specific context. For example, the personal home page of an individual would have normally been created within the context of that person's affiliation and organizational role. Some readers might be very familiar with such context, while others might not. In the latter case, the use of Magpie is especially beneficial, given that the context would be made explicit to the reader and context-specific functionalities will be provided. Because different readers show differing familiarity with the information shown in a web page and with the relevant background domain, they require different level of sensemaking support. Hence, the semantic layers in Magpie are designed with specific types of user in mind.

The semantic capabilities of Magpie are achieved by creating a *semantic layer* over a standard HTML web page. The layer is based on a particular ontology selected by the user, and associated *semantic services*. In the context of our paper, the services are defined separately from the ontology, and are loosely linked to the ontological

hierarchy. This enables Magpie to provide different services depending on the type of a particular semantic entity that occurs in the text. In addition to this shallow semantics, one of the key contributions of the Magpie architecture is its ability to facilitate *bi-directional communication* between the client and server/service provider. This is achieved through so-called *trigger services*, which may feature complex reasoning using semantic entities from the user-browsed pages as data source. Triggers use ontology to recognize interesting data patterns in the discovered entities, and bring forward semantically related information to the user. The key benefit of this approach is that there may be no explicit relationship expressed in the web page – the relevance is established implicitly by consulting a particular ontology.

Magpie is an example of collaboration within AKT leading to new opportunities. One of the early collaborations within AKT (called AKT-0) combined dynamic ontologically based hyperlink technology from Southampton with the OU's own ontology-based technologies. The final result of the AKT-0 collaboration was an extended Mozilla browser where web pages could be annotated on-the-fly with an ontology generated lexicon facilitating the invocation of knowledge services.

6.3. Identifying communities of practice

Communities of practice (COPs) are informal self-organising groups of individuals interested in a particular practice. Membership is not often conscious; members will typically swap war stories, insights or advice on particular problems or tasks connected with the practice (Wenger 1998). COPs are very important in organisations; taking on important knowledge management functions. They act as corporate memories, transfer best practice, provide mechanisms for situated learning, and act as foci for innovation.

Identifying COPs is often regarded as an essential first step towards understanding the knowledge resources of an organisation. COP identification is currently a resource-heavy process largely based on interviews that can be very expensive and time consuming, especially if the organisation is large or distributed.

ONTOCOPI (Ontology-based Community of Practice Identifier, http://www.aktors.org/technologies/ontocopi/) attempts to uncover COPs by applying a set of ontology network analysis techniques that examine the connectivity of instances in the knowledge base with respect to type, density, and weight of these connections (Alani et al 2003a). The advantage of using an ontology to analyse such networks is that relations have semantics or types. Hence certain relations – the ones relevant to the COP – can be favoured in the analysis process.

ONTOCOPI applies an expansion algorithm that generates the COP of a selected instance (could be any type of object, e.g. a person, a conference) by identifying the set of close instances and ranking them according to the weights of their relations. It applies a breadth-first, spreading activation search, traversing the semantic relations between instances until a defined threshold is reached. The output of ONTOCOPI is a ranked list of objects that share some features with the selected instance.

COPs are often dynamic – one typically moves in different communities as one's working patterns, seniority, etc, change in the course of one's career. If temporal information is available within the ontology being analysed, then ONTOCOPI can use it to present a more dynamic picture. For example, when an ontology is extended to allow representation of the start and end dates of one's employment on a project, it is then possible to exploit that information. ONTOCOPI can be set to focus only on

relationships obtained within some specified pair of dates, ignoring those that fall outside the date range.

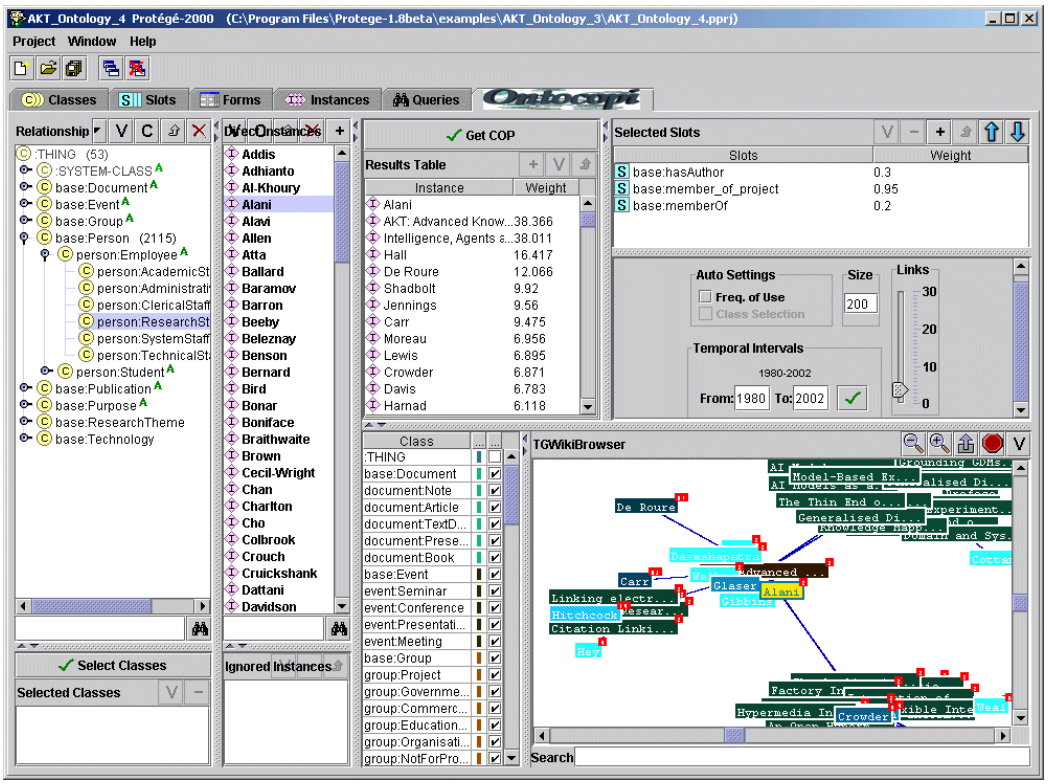


Figure 12 The Protégé Version of the COP technology

ONTOCOPI currently exists in three different implementations; a plugin to Protégé 2000 (http://protege.stanford.edu/); an applet working with the triplestore (section 9.2); and as URI query to the 3store that returns COPs in RDF.

COP detection is an application of the basic technique of ontology-based network analysis, and this general technique of knowledge retrieval can play an indirect role in a number of other management processes. In AKT, we have applied ONTOCOPI's analysis to bootstrap other applications, such as organisational memory (Kalfoglou et al 2002), recommender systems (Middleton et al 2002), and ontology referential integrity management system (Alani et al 2002 – see section 8.4).

7. Publishing

Knowledge is only effective if it is delivered in the right form, at the right place, to the right person at the right time. Knowledge publishing is the process that allows getting knowledge to the people who need it in a form that they can use. As a matter of fact, different users need to see knowledge presented and visualised in quite different ways. Research on personalised presentations has been carried out in the fields of hypermedia, natural language generation, user modelling, and human-computer interaction. The main challenges addressed in AKT in these areas were the connection of these approaches to the ontologies and reasoning services, including modelling of user preferences and perspectives on the domain.

Research on knowledge publishing in AKT has focused on three main areas:

- CS Active Space – intelligent user-friendly knowledge exploration without complex formal queries
- Artequakt – personalised summaries using story templates and adaptive hypermedia techniques
- MIAKT-NLG – natural language generation (NLG) of annotated images and personalised explanations from ontologies

CS AKTive Space (section 10.1) is an effort to address the problem of rich exploration of the domain modelled by an ontology. We use visualization and information manipulation techniques developed in a project called mSpace (schraefel et al 2003) first to give users an overview of the space itself, and then to let them manipulate this space in a way that is meaningful to them. So for instance one user may be interested in knowing what regions of the country have the highest level of funding and in what research area. Another user may be interested in who the top CS researchers are in the country. Another might be interested in who the up and comers are and whether they're all going to the same university. CS AKTive space affords just these kinds of queries through formal modelling of information representation that goes beyond simple direct queries of an ontology and into rich, layered queries (Shadbolt et al 2003).

We have mentioned Artequakt already (section 3.3), in the context of acquisition. As far as publishing goes, the Artequakt biography generator is based around an adaptive virtual document that is stored in the Auld Linky contextual structure server. The virtual document acts as a story template that contains queries back into the knowledge-base. Each query resolves into a chunk of content, either by retrieving a whole sentence or paragraph that contains the desired facts or by inserting facts directly from the knowledge-base into pre-written sentence templates. It is possible to retrieve the story template in different contexts and therefore get different views of the structure and in this way the story can be personalised. The contribution of the Artequakt system is in the ontological approach to the extraction, structuring and storing of the source texts and in the use of adaptive virtual documents as story templates.

Using whole fragments of pre-written text is surprisingly effective, as a reader is very forgiving to small inconsistencies between adjacent fragments. However, these fragments already contain elements of discourse that might be inappropriate in their new context (such as co-referencing and other textual deixis) and which can prove jarring for a reader. We are now starting to explore the use of NLG techniques for the MIAKT application (section 10.3). There are two anticipated roles; (i) taking images annotated with features from the medical ontology and generating a short natural language summary of what is in the image, (ii) taking medical reports and personalising them – for example removing technical or distressing terms so that a patient may view the records or else anonymising the report so that the information may be used in other contexts.

In addition to personalised presentation of knowledge, NLG tools are needed in knowledge publishing in order to automate the ontology documentation process. This is an important problem, because knowledge is dynamic and is updated frequently. Consequently, the accompanying documentation which is vital for the understanding and successful use of the acquired knowledge, needs to be updated in sync. The use of NLG simplifies the ontology maintenance and update tasks, so that the knowledge

engineer can concentrate on the knowledge itself, because the documentation is automatically updated as the ontology changes. The NLG-based knowledge publishing tools (MIAKT-NLG) also utilise the ontology instances extracted from documents using the AKT IE approaches (see section 6.1). The dynamically generated documentation not only can include these instances, as soon as they get extracted, but it can also provide examples of their occurrence in the documents, thus facilitating users' understanding and use of the ontology. The MIAKT-NLG tools incorporates a language generation component, a Web-based presentation service, and a powerful user-modelling framework, which is used to tailor the explanations to the user's knowledge, task, and preferences.

The challenge for the second half of the project in NLG for knowledge publishing is to develop tools and techniques that will enable knowledge engineers, instead of linguists, to create and customise the linguistic resources (e.g., domain lexicon) at the same time as they create and edit the ontology (Bontcheva et al 2001, Bontcheva 2001).

8. Maintenance

Knowledge maintenance is key to retaining the value of a repository of knowledge (O'Hara & Shadbolt 2001); knowledge can quickly date, or become radically and immutably decontextualised by a rapidly changing environment. Inconsistencies will inevitably arise over time, particularly in an additive medium such as the WWW, where there are many more incentives to post some information than to withdraw it.

So far, AKT has focused on three major aspects of knowledge maintenance, the issues of situating knowledge with respect to changing ontologies (section 8.2), the problem of establishing and maintaining mappings between ontologies (section 8.3), and of attempting to deal with the pernicious problem of deciding when two distinct terms refer to the same object (section 8.4). AKT has also examined, in the context of information overload, the notion of forgetting. Although work here is at a preliminary stage, this is likely to become a "hotter topic" in computer science over the near future. Hence we will begin with a brief discussion of forgetting.

8.1. Forgetting

AKT researchers have been examining the notion of forgetting, in both human and machine memory, to try to understand the processes involved; although increases in computer storage capacity mean that forgetting is no longer needed for reasons of technical efficiency, as a knowledge maintenance strategy it shows much promise (O'Hara et al 2003). As external digital devices become increasingly essential for access to information, and information accumulates, demands on search engines and other retrieval systems will become even greater; in such an environment, despite the massively increased storage capacity of systems, the demands for near-instant access to information may mean that forgetting is a sensible strategy.

Of course, this intuition may be false; artful organisation of knowledge repositories may be sufficient to enable efficient search in most contexts. However, there may be a role for forgetting even with respect to such a strategy; one way of organising a repository would be to foreground information that can heuristically be tagged as being likely to be accessed, or current, and to place in the background information that may be out of date, or merely irrelevant. Such backgrounded information would then be harder to get at, as the search strategy would focus on the foregrounded

information first. In a certain sense, then, the backgrounded information can be seen as "forgotten"; at any rate, the backgrounded information would clearly overlap with the information that would be genuinely forgotten under a strategy of deletion.

The human memory system has evolved in such a way that forgetting garbage is a central strategy in its efficiency (Schacter 2001). It therefore makes a great deal of sense to try to look at human memory as a way of establishing useful metaphors for computing research, or indeed to try to reconceptualise computing strategies in terms of human memory (O'Hara et al 2000, O'Hara et al 2003). For example, evidence suggests dreaming in humans seems to facilitate the process of memorising and forgetting by allowing a reorganisation of memory in the absence of input from perception; in certain ways, this might be seen as analogous to garbage collection; such analogies may help improve computer memory management.

8.2. Ontology change

As the end user's requirements change over a period of time, the ontologies used to structure and inform knowledge assets may evolve to better suit their requirements. This gradual drift in the ontologies may lead to issues of legacy knowledge which is now being effectively expressed using a different language to that in common use.

The ConRef system (Winter et al 1998) is used to manage service or object inventories that use more than one ontology. It provides a service that transparently maps the queries expressed in the user's ontology into the ontology used by the inventory, and allows the user to modify the mapped query by stipulating which of the returned objects are unwanted, or those objects that should have been returned. ConRef is currently being used in a prototype system with British Aerospace for managing an inventory of fasteners.

8.3. Ontology mapping

The more ontologies are being deployed on the SW, the greater the demand to share them for the benefits of knowledge sharing and semantic interoperability. The sharing of ontologies though, is not a solved problem. It has been acknowledged and researched by the knowledge engineering community for years.

One aspect of ontology sharing is to perform some sort of mapping between ontology constructs. That is, given two ontologies, one should be able to map concepts in one ontology onto those in the other. Further, research suggests that we should also be able to combine ontologies where the product of this combination will be, at the very least, the intersection of the two given ontologies. These are the dominant approaches that have been studied and applied in a variety of systems (Kalfoglou & Schorlemmer 2003a).

There are, however, some drawbacks that prevent engineers from benefiting from such systems. Firstly, the assumptions made in devising ontology mappings and in combining ontologies are not always exposed to the community and no technical details are disclosed. Secondly, the systems that perform ontology mapping are often either embedded in an integrated environment for ontology editing or are attached to a specific formalism. Thirdly, in most cases mapping and merging are based on heuristics that mostly use syntactic clues to determine correspondence or equivalence between ontology concepts, but rarely use the meaning of those concepts, i.e., their semantics. Fourthly, most, if not all approaches do not exploit ontological axioms or rules often found in formal ontologies. Finally, ontology mapping as a term has a

different meaning in different contexts due to the lack of a formal account of what ontology mapping is. There is an observed lack of theory behind most of the works in this area (Kalfoglou & Schorlemmer 2003a).

Motivated by these drawbacks we have developed a method and a theory for ontology mapping and merging. The approach draws heavily on proven theoretical work but our work goes further in providing a systematic approach for ontology mapping with precise methodological steps. In particular, our method, Information-Flow based Ontology Mapping (IF-Map) (Kalfoglou & Schorlemmer 2003b), draws on the proven theoretical ground of Information Flow and channel theory (Barwise & Seligman 1997), and provides a systematic and mechanised way for deploying the approach in a distributed environment to perform ontology mapping among a variety of different ontologies.

The IF-Map system formalizes mappings of ontology constructs in terms of logic infomorphisms, the fundamental ingredient of Information Flow. These are well suited for representing the bi-directional relation of types and tokens, which corresponds to concepts and instances in the ontology realm. IF-Map is focusing on instances and how these are classified against ontology concepts. This reveals the operational semantics that the ontology's community has chosen by virtue of how it uses its instances. The IF-Map algorithm makes use of this information in order to map onto related concepts from another ontology with which its concepts classify the same instances.

The methodological part of IF-Map consists of four major steps: (a) ontology harvesting, (b) translation, (c) infomorphism generation, and (d) display of results. In the ontology-harvesting step, ontology acquisition is performed. We use a variety of methods: use existing ontologies, download them from publicly accessible online ontology libraries, edit them in ontology editors, or harvest them from the Web (section 3.1). This versatile ontology acquisition step results in a variety of ontology language formats, ranging from KIF and Ontolingua to OCML, RDF, Prolog, and native Protégé knowledge bases. This introduces the second step, that of translation. Although there are a wide choice of translators in the public domain, we found it practical to write our own translators. We did that to have a partial translation, customised for the purposes of ontology mapping in terms of IF-Map where the only constructs needed are the concepts and a representative instance for each one of them. The next step is the main mapping mechanism: the IF-Map algorithm. We provide a Java front-end to the Prolog-written IF-Map program so that it can be accessed from the Web, and we also provide a Java API to enable external calls to IF-Map from other systems. This step will find infomorphisms, if any, between the two ontologies under examination, and in the last step of the process we display them in RDF format. Finally, we also store the results in a knowledge base for future reference and maintenance reasons.

We have used IF-Map with success in a variety of ontology mapping scenarios within and outside AKT such as mapping of computer science departments' ontologies from AKT participating universities (Kalfoglou & Schorlemmer 2002); mapping of e-government ontologies from a case study using UK and US governments ministries (Schorlemmer & Kalfoglou 2003). We have also conducted a large-scale survey of the state-of-the-art of ontology mapping (Kalfoglou & Schorlemmer 2003a) and we are currently exploring the role of Information Flow and the IF-Map approach in the

wider context of semantic interoperability and integration (Schorlemmer & Kalfoglou 2003 – submitted).

8.4. Coreference resolution

The acquisition of knowledge from heterogeneous sources by automatic means (as by the AKT harvesters as part of Hyphen – Shadbolt et al 2003) carries with it certain problems, of which unintentional coreference is a key issue; different sources may refer to the same entities by different means. If we are to be able to reason effectively with this knowledge, we need to be able to determine which entities are coreferent, and to collapse these multiple instances into a canonical representation.

We have developed tools for the identification and resolution of coreferent entities in a knowledge base (Alani et al 2002). An initial step is to perform some kind of clustering in order to create a smaller "window" over which pairwise comparisons are made in order to ascertain possible duplication. Elements in each pair in these windows are identified to be in some relation based on the attributes recorded in the ontology. These (often overlapping) clusters are then subjected to more computationally intensive methods until disjoint sets of co-referential identifiers are obtained. This is achieved by establishing that there is an equivalence relation between elements of a (possibly) coreferential set which we then quotient the set of identifiers by. The relations we build upon are comparisons of features based on attributes of entities (such as e-mail address) and distances based on string matching algorithms. We also include derived features obtained by composing ontological relations, in particular, the network analyses that underlie the community of practice identification methods (Alani et al 2003a). We have employed a number of machine learning methods to identify the clusters and performed checks for the transitivity of these identification relations in order to determine whether we obtain equivalence relations.

We have taken a lighter approach based on heuristic rules that are consistent with the more comprehensive methods just described for the construction of a knowledge base describing UK HE computer science research which has been subsequently used in the CS AKTive Space demonstrator (Shadbolt et al 2003, and section 10.1 below).

9. Infrastructure

Integration is a key aspect to the AKT approach, given the fissiparous tendency of any project that is based over several discrete lifecycle steps. However, the SW is a fast-moving domain for which integration via an overarching and monolithic infrastructure would be a mistake; it is important to produce lightweight infrastructures for different problem classes that can underpin opportunistic combinations of heterogeneous services – some or all of which may originate outside AKT.

AKT's infrastructure focus has been varied through its history. In this section, we will focus on four areas: the AKTbus knowledge interchange mechanism, the AKT Triplestore, an experiment in large-scale service curation techniques, the development of Internet Reasoning Services, and Human Language Technology infrastructure.

9.1. AKTbus

The AKTbus was designed as a lightweight XML-based messaging infrastructure, to integrate a number of pre-existing systems and components from the consortium members. The AKTbus is designed to carry a variety of content languages, including

simple query-response protocols such as OKBC, and agent communication languages such as KQML and FIPA-ACL. At the present time, the AKTbus infrastructure has the following components:

- an XML-based messaging protocol;
- reference implementations (fully interoperable) and APIs for Java and Prolog.

The AKTbus can be viewed as a "simpler SOAP", as it shares many of the aims and design features of the W3C's Simple Object Access Protocol; however, when work on AKT began, the SOAP specification was in a state of flux, and implementations were primitive and unstable. Moreover, SOAP is primarily geared to support synchronous remote procedure calls, while the AKTbus is primarily an asynchronous message-passing protocol, better-suited to communication among autonomous knowledge-driven systems, such as software agents.

This infrastructure has been used to build a number of proof-of-concept knowledge systems integrations:

specifications and implementations of FIPA and KQML agent communication languages running over the AKTbus protocol, providing a gateway to the *FIPA-based Agentcities network*, and the *KQML-based KRAFT agent system* (Hui et al, 2003);

- an AKTbus interface to the *OKBC-based Protégé server*;
- an AKTbus interface to Edinburgh's I-X components, including their *process panels* and *broker.*

The AKTbus also, as we have noted, underpins the KRAFT/I-X TIE (section 5.1).

9.2. The AKT ontology and triplestore

The CS AKTiveSpace testbed (discussed further in Section 10.1) requires a dataset that describes computer science research in UK higher education. The AKT Reference Ontology was written to provide principled structure to this data, focusing on the application domain of UK academic life. Its construction has been informed by other work on the design of moderate scale ontologies, from the Open University's experiences with OCML, to external efforts such as IEEE SUO and the Cyc project.

The design and maintenance of this ontology has been a collaborative endeavour involving all AKT partners, with regular sessions at project workshops to review the performance and suitability of the ontology. The reference ontology is in many ways a living artifact, and one which we are using to explore the issues of ontology evolution and maintenance. To ensure maximal interoperability with the different tools in use in the consortium, the ontology is maintained in five different languages of varying degrees of expressivity: OCML, OntoLingua, DAML+OIL, OWL and RDF Schema.

The AKT triple store was developed to provide a storage and query interface to the large volumes of RDF data that the project requires to research the possibilities of a semantically described domain. The triple store was designed to take advantage of established relational database technologies and techniques to allow for efficient access to the RDF data and entailments. The principles behind this technique have been described in (Harris and Gibbins 2003). The software, 3store, has been tested on knowledge bases with up to 25 million RDF triples and is expected to scale to much larger volumes, which makes it one of the most scalable generic RDF storage technologies currently available. It has been released under a Free Software licence to

enable its use in other research projects and by commercial entities and has been downloaded by around 70 individuals and projects. A number of AKT systems have been connected to the triple store such as the I-X Process Panels and on a larger scale the CS AKTive Space IFD described in Section 10.1.

The AKT triple-store system (3store) is written in C for POSIX compliant systems, and so is portable to most UNIXes. It uses MySQL as its back-end repository, storing RDF triples in an ontology neutral database schema. It is available from SourceForge at http://www.sourceforge.net/projects/threestore/. RDFS entailments are generated using a hybrid eager/lazy approach (Harris & Gibbins 2003), where queries are adaptively translated into SQL database queries and executed by the RDBMS engine. This allows for very efficient queries (typically a few milliseconds) over large knowledge bases.

9.3. Internet Reasoning Services

As we have argued, SW services hold enormous potential. The augmentation of web services with formal descriptions of their competence will facilitate their automatic location, mediation, and composition. The IRS-II (Internet Reasoning Service) is a framework and implemented infrastructure to support the publication, location, composition and execution of heterogeneous web services, augmented with semantic descriptions of their functionalities (Motta et al 2003). IRS-II has three main classes of features which distinguish it from other work on semantic web services. Firstly, by automatically creating wrappers, standalone software (we currently support Java and Lisp) can be published through the IRS-II very easily. Secondly, because IRS-II is built on a knowledge modelling framework, we provide support for *capability-driven* service invocation, for flexible mappings between services and problem specifications and we support dynamic, knowledge-based service selection. Finally, IRS-II services are web service compatible – standard web services can be trivially published through the IRS-II and any IRS-II service automatically appears as a standard web service to other web service infrastructures.

The approach taken in IRS-II is based on the UPML framework (Fensel and Motta, 2001), one of the main results of the EU funded IBROW project (Benjamins et al., 1998). The UPML framework partitions knowledge into domain models, task models, and problem solving methods each supported by appropriate ontologies. Domain models capture the essential concepts and relationships in a domain. Task models contain declarative representations of capabilities, specifically, the goal of a task, the types of inputs and outputs, and pre-conditions. Problem solving methods can be thought of as knowledge level descriptions of generic reasoners which can be harnessed to solve tasks. One of our main contributions in this work has been to integrate the UPML framework with web services.

9.4. Human language technology infrastructure

Human Language Technology (HLT) plays an increasingly important role in KM in the context of the Semantic Web. With the expansion of KM, and the application of knowledge technologies, to large-scale tasks in the SW or corporate intranets, so the need for robust, large-scale HLT is increased. Our work in this area focuses on supporting experimental repeatability and quantitative measurement, within an open source environment, GATE (Cunningham et al 2002), engineered to an exceptional standard. GATE has thousands of users at hundreds of sites.

As we have seen, ontologies and reasoning abilities are necessary components across systems to exploit the possibilities of the SW, certainly for most HLT applications. Such ontologies may be created by people or applications independent of the HLT context; perhaps by merging old ontologies, or by automatic generation of ontologies from some legacy sources. Since such ontologies may be created independently of an HLT application, their suitability may be marginal; and given the lack of standards it is likely that many ontologies will contain hard-to-detect errors. HLT components – which often do not support ontologies and reasoning as explicitly reusable components – will often need to exchange knowledge with other services and technologies that provide non-HLT services.

Hence there are clear opportunities for HLT infrastructures within AKT to exploit. There are requirements for HLT infrastructures that support ontologies and reasoning, particularly supporting existing ontology standards such as DAML+OIL. Ontology maintenance and use should become part of the HLT application building process, and ontology-aware HLT components should become capable of sharing and exchanging ontologies with other tools, including non-HLT tools.

To this end, AKT has extended the world-leading GATE infrastructure to support SW-enabled HLT, providing interoperability with existing SW tools and other knowledge technologies (including from within the AKT consortium, of course) via standards such as RDF and DAML+OIL. Integration of linguistic data and knowledge is being achieved via ontologies, and there is support for the creation of ontology-aware HLT modules. Ontology API allows unified access by HLT components, regardless of the original ontology format, making it easier for components to exchange ontologies, following a similar route to the exchange of standard linguistic data (e.g. lexicons). AKT HLT infrastructures now allow integration with knowledge technologies, and therefore their creative reuse; for example, Protégé has been used for ontology visualisation and maintenance.

9.5. Collaborative Work Environments

We have invested significant effort in integrating existing and developing new tools for collaborative work. Moreover, as demonstrated in the Management Report we have used these tools for organising and supporting our own management and work processes.

9.5.1 CoAKTinG Collaborative Technologies

The CoAKTinG testbed (Collaborative Advanced Knowledge Technologies in the Grid: www.aktors.org/coakting) is extending and integrating AKT technologies specifically to support distributed scientific collaboration – both synchronous and asynchronous – over the Grid and standard internet. Now halfway through, the CoAKTinG testbed has integrated technologies for instant messaging/presence awareness (OU), real time decision rationale and group memory capture (OU), issue handling and coordination support (Edinburgh), and semantically annotated audio-visual streams (Southampton). See the summary in Table 1; details of the approaches plus an extended use scenario can be found in Buckingham Shum, et al. (2002).

Pair-wise integrations between the above tools has been demonstrated, while an example of multi-way integration is the meeting navigation and replay tool (Bachler, et al., 2003) illustrated in Figure 13, which integrates metadata grounded in a meeting ontology for scientific collaboration, with time-based metadata indicating current

slide, speaker, and issue under discussion, to enable novel forms of meeting navigation and replay.

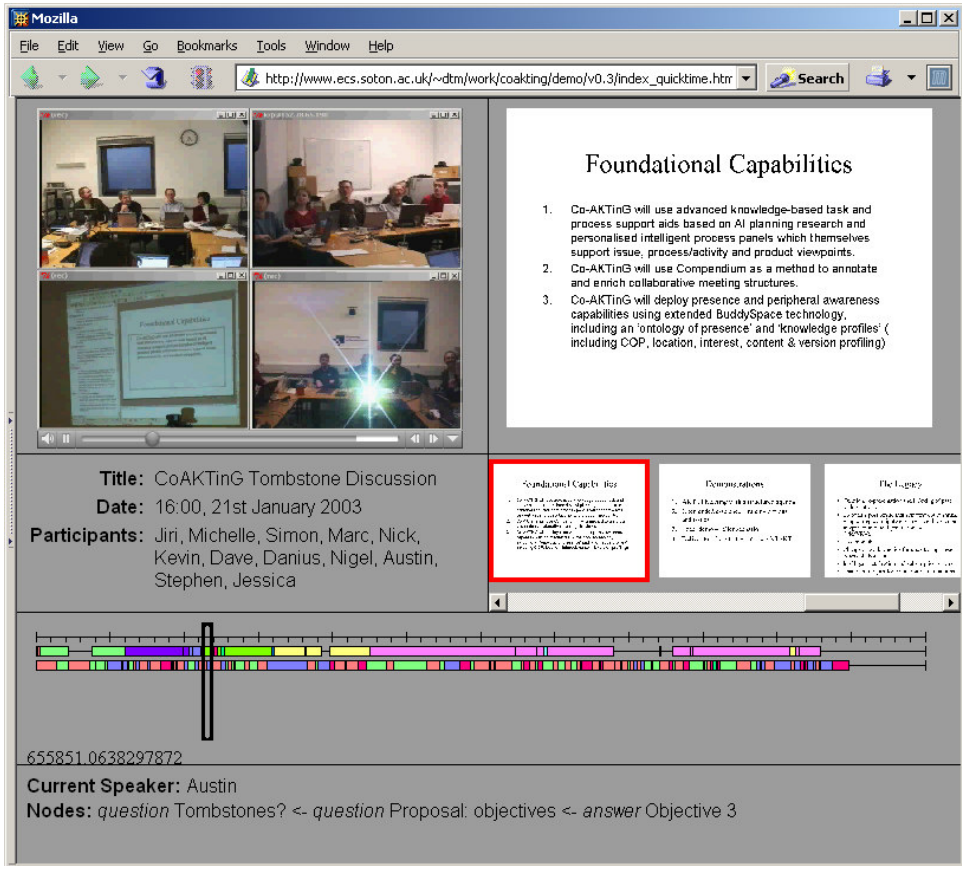


Figure 13: CoAKTinG proof of concept web interface for navigating an AV meeting record by time, speaker, slide, and issue under discussion. The two coloured bars indicate slide transitions (top) and speaker (bottom). The current speaker at the time selected is indicated at the foot, plus an indication of the current issue under discussion, extracted from the Compendium record (Bachler, et al., 2003).

CoAKTinG Collaboration Technologies
Ontologically annotated audio/video streams for meeting navigation and replay. Few researchers have the time to sit and watch videos of meetings; an AV record of an online meeting is thus only as useful as its indexing. Moreover, indexing effort must negotiate the cost/benefit tradeoff or it will not be done. Prior work has developed ways to embed 'continuous metadata' (of which one form is hyperlinks) in media streams. Within CoAKTinG, this work forms the platform for integrating multiple metadata streams with AV meeting records (as illustrated in the meeting replay tool).
Issue handling, tasking, planning and coordination. We are building applications using I-X Intelligent Process Panels and their underlying <I-N-C-A> (Issues, Nodes, Constraints and Annotations) constraint-based ontology for processes and products [i-x.info]. The process panels provide a simple interface that acts as an intelligent "to do" list that is based on the handling of issues, the performance of activity or the addition of constraints. It also supports semantically task directed "augmented" messaging and reporting between panel users. A common ontology of processes and process or collaboration products based on constraints on the collaborative activity or on the alternative products being created via the collaboration is the heart of this research.
Collective sensemaking and group memory capture. Whilst meetings are a pervasive knowledge-based activity in scientific life, they are also one of the hardest to do well. "Meeting technologies" tend either to over-structure meetings (e.g. Group Decision Support Systems), or ignore process altogether, and simply digitize physical media (e.g. whiteboards) for capturing the products of discussion. The *Compendium* approach [www.CompendiumInstitute.org] occupies the hybrid middle-ground – 'lightweight' discussion structuring and mediation plus idea capture, with import and export to other document types. Dialogue maps are created on the fly in meetings providing a visual trace of issues, ideas, arguments and decisions.
Enhanced presence management and visualisation. The concept of *presence* has moved beyond the 'online/offline/away/busy/do-not-disturb' set of simple state indicators towards a rich blend of attributes that can be used to characterise an individual's physical and/or spatial location, work trajectory, time frame of reference, mood, goals, and intentions. Our challenge is how best to characterise presence, how to make it easy to manage and easy to visualise, and how to remain consistent with the user's own expectations, work habits, and existing patterns of Instant Messaging (IM) and other communication tool usage. Working with the Jabber open source XML-based communications architecture, we have released a prototype called *BuddySpace* which overlays presence information onto visualisations, both geographical (e.g. a map of a building, or a region), and conceptual (e.g. a workflow chart or project plan, a design or experiment). [kmi.open.ac.uk/projects/buddyspace]

Table 1: CoAKTinG Collaboration Technologies

9.5.2 Use Case: Building the AKT Reference Ontology

An example of our use of our own collaborative work environments can be seen in the production of the AKT reference ontology is shown in Figure 14. The life-cycle consists of a pre-meeting, a live meeting and a post-meeting stage. The final outcome is the AKT Reference ontology published in a variety of formats and representation languages.

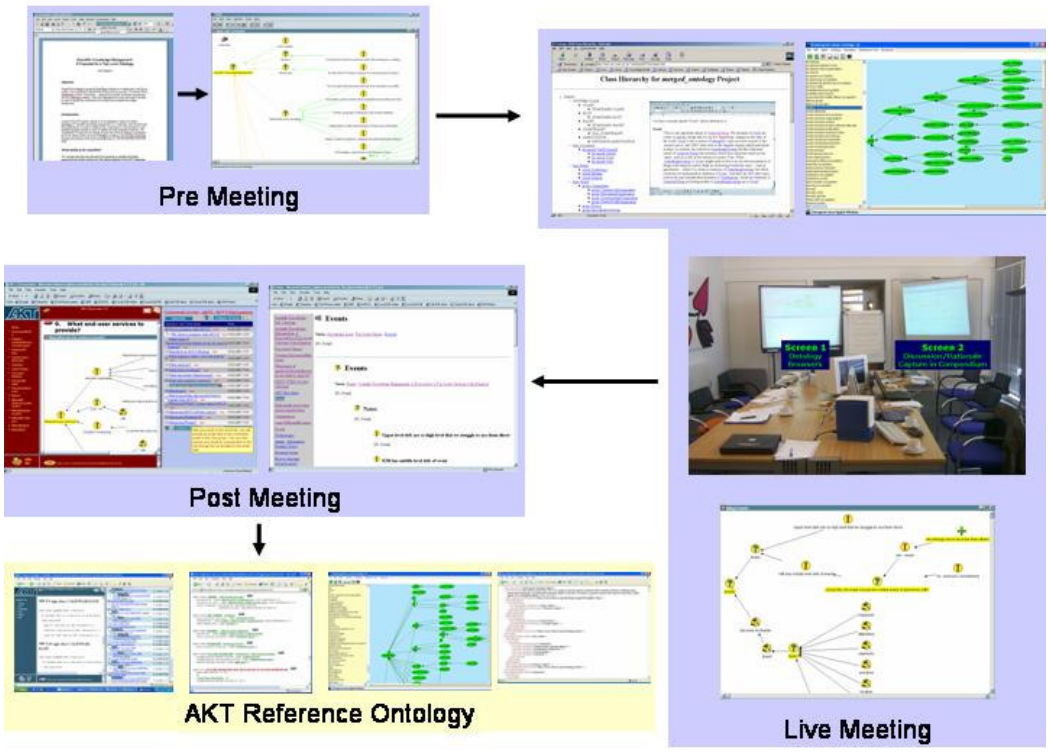


Figure 14 The life-cycle used to develop the AKT reference ontology

In the pre-meeting stage a proposal document written by a researcher at Edinburgh was automatically transformed into a Compendium map. The headings were turned into questions and the first line of each paragraph was turned into a statement.

The setup for the live meeting is shown in Figure 15. Participants sat around a table in a room containing two screens. The first displayed candidate ontology definitions held on various ontology servers. As the meeting progressed the discussion was captured in a Compendium map – see the screen snapshot in the bottom right of Figure 14.

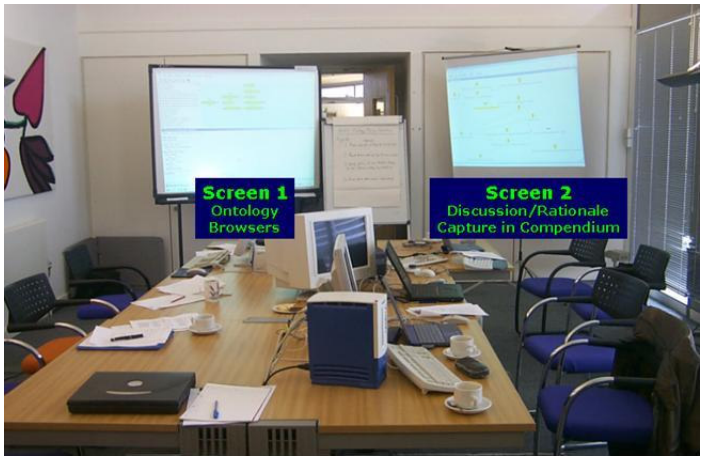


Figure 15 Room Layout for AKT Reference Ontology Meeting

Once the meeting had finished the Compendium map was exported both as a series of images (see the screen snapshot on the middle far left of Figure 14) and as a HTML page (the screen snapshot on the middle of Figure 14). Using D3E (see

http://www.aktors.org/technologies/D3E - Digital Document Discourse Environment/) the exported maps were integrated into a threaded discussion space. The D3E toolkit is able to automatically create and link discussion threads to plain HTML documents. New comments were automatically sent to an AKT Reference Ontology mailing list.

Using the discussion site the main issues raised in the meeting were considered for a further month. At the end of the month an initial version of the ontology was codified in OCML (Motta, 1999) and published using the D3E toolkit. Once all comments on the initial version were collected a second was created and released in a variety of formats (see Figure 14 bottom left) including OCML (as a hyperlinked set of web pages and on the WebOnto server), Ontolingua (a hyperlinked set of web pages), DAML+OIL and OWL. All of these versions are available from www.aktors.org/publications/ontology/.

There are two main lessons can we draw from our experiences of technology support of collaborative work within AKT. Firstly, a blended model of collaboration using a variety of technologies is required to support technical work. At the start of a collaborative venture, whilst the specific problem is being framed face-to-face meetings are necessary. Later on synchronous virtual meetings and asynchronous communication, via D3E and email, can be used to co-ordinate evolutionary technical development. From time-to-time further face-to-face meetings are required to re-frame the problem or to unify significant developments. This model bears some resemblance to Fischer's Seeding Evolutionary Growth and Reseeding model (Fischer et al., 1994) for the incremental development of design systems.

The second lesson is based on the way documents are used within organisations. Rather than serving as a means to objectively transmit knowledge the key role of documents within organisational settings is to support negotiation and interpretation as communities struggle to reach a shared understanding. Building on the OU's Enrich methodology and Southampton's rich linking technology (Motta et al., 2000; Carr et al 2001), the key to sharing knowledge within AKT has been to support the enrichment of our collaborative work representations.

10. Applications

The AKT concept was always based strongly around the idea of testbeds. Such testbeds – real-world contexts for the application of knowledge services and technologies – are important not only for robust proof of concept, but also to aid the process of integration of the AKT components and the transfer of results to the user community. We present three under development at present.

10.1. CS AKTive Space

CS AKTive Space attempts to provide an overview of current UK University based research in Computer Science. The application exploits a wide range of semantically heterogeneous and distributed content relating to Computer Science research in the UK. It provides services such as browsing topics and institutions for researchers, it can show the geographic range and extent of where a topic is researched, provides an estimation of "top" researchers in a topic and by geographic region, is able to calculate a researcher's Community of Practice. We chose this area for a number of reasons; (i) we had a real interest in having such a set of services, (ii) it is a domain that we understand, (iii) it is relatively accessible and easy to communicate as a

domain, (iv) we were able to secure access to a wide range of content that was not subject to industrial embargo, (v) it presented real challenges of scale and scope.

The application exploits a wide range of semantically heterogeneous and distributed content relating to Computer Science research in the UK. For example, there are almost 2000 research active Computer Science faculty, there are 24,000 research projects represented, many thousands of papers, hundreds of distinct research groups.

This content is gathered on a continuous basis using a variety of methods including harvesting and scraping (Leonard and Glaser 2001) as well as other models for content acquisition. The content currently comprises around seven million RDF triples and we have developed storage, retrieval and maintenance methods to support its management (Harris and Gibbins, 2003). The content is mediated through an ontology (http://www.aktors.org/publications/ontology/) constructed for the application domain and incorporates components from other published ontologies (Niles and Pease, 2001).

CS AKTive Space supports the exploration of patterns and implications inherent in the content. It exploits a variety of visualisations and multi dimensional representations that are designed to make content exploration, navigation and appreciation direct and intuitive (schraefel et al 2003). As mentioned the knowledge services supported in the application include investigating communities of practice (Alani et al 2003a) and scholarly impact (Kampa 2002).

We aim to provide a content space in which a user can rapidly get a Gestalt of who is doing what and where, what are the significant areas of effort both in terms of topic and institutional location, what of this work is having an impact or influencing others and where are the gaps in research coverage. In Figure 16 we see a screenshot of CS AKTive Space. We can see that on the right a region has been selected in the middle of the country by dragging a reticule of selected radius over it. This action will have dispatched a complex query to our underlying RDF repository the results being all the Computer Science Departments in that region and the corresponding topics that they research according to a taxonomy of research taken from the ACM. In the case of Figure 16 we have selected *artificial intelligence* as the topic of interest and a further compound query will have been sent to return in this case the top 5 researchers in the area. This is currently determined by impact factors such as the size of their grant portfolio but we will be making this much more customisable in future versions.

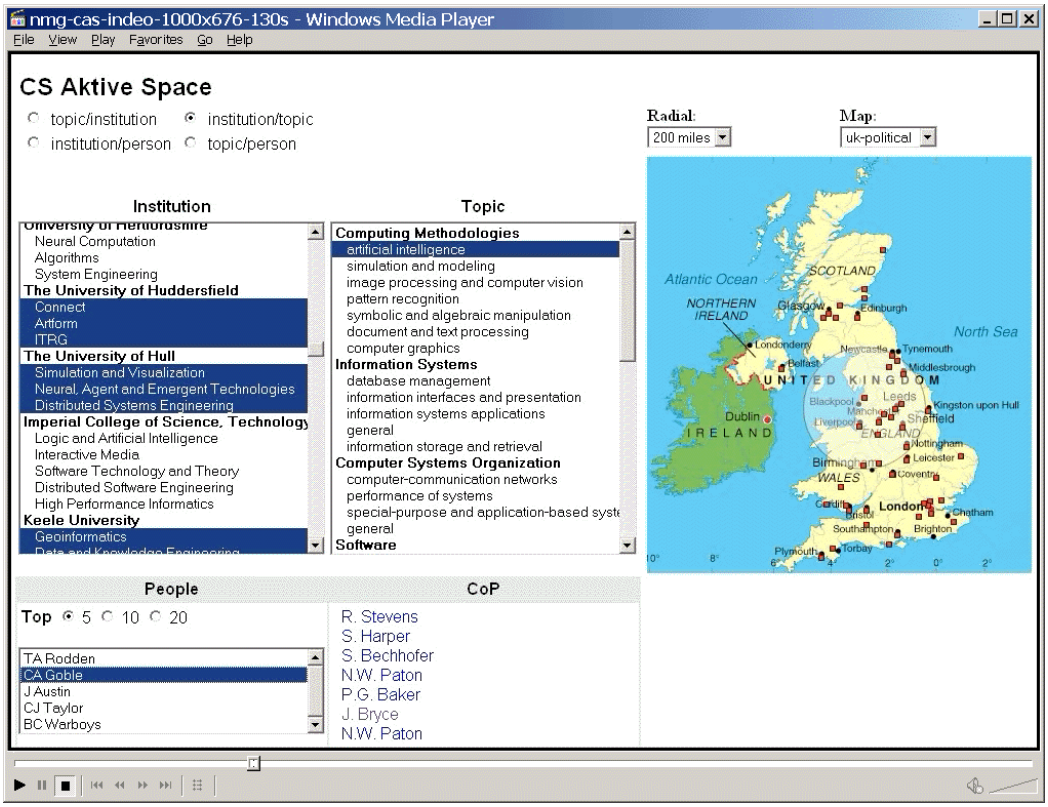


Figure 16 CS AKTive Space: A Semantic Web Application for UK Computer Science Research

Finally one of the researchers has been selected *Goble* and her community of practice has been calculated – showing us those researchers with whom she has strong links in terms of relations such as co-authoring, co-investigator and so on.

This work illustrates a number of substantial challenges for the Semantic Web. There are issues to do with how to best sustain an acquisition and harvesting activity. There are decisions about how best to model the harvested content; how to cope with the fact that there are bound to be large numbers of duplicate items that need to be recognised as referring to the same objects or referents; the degree to which our inferential services can cope as more content becomes available; how we present the content so that inherent patterns and trends can be directly discerned must be considered; how trustworthy is the provenance and accuracy of the content; and how all this information is to be maintained and sustained as a social and community exercise.

Finally, it is worth making a brief note here about our approach to visualisation of research issues. The key activity of a researcher is research: the systematic investigation of a corpus of literature. Digital libraries provide integrated browse-and-search access to large collections of scientific and technical literature, but many of the the questions which researchers bring to a digital library (such as “What other papers did this project publish?” or “What are the significant research groups in this field?”) are situated in the knowledge domain and not supported by current Web environments (Kampa 2002).

By applying (Figure 17) the AKT ontology and triple store to Southampton’s existing OpCit bibliographic analysis environment (Hitchcock et al 2002), we have produced an infrastructure that offers a portfolio of knowledge services that improve digital

library browsing by offering community analyses (Figure 18) as well as co-citation visualisation (Figure 19). To increase the range of visualisation types available, an independent geographic visualisation interface to the Triple Store was developed (Figure 20). This, as we have seen (Figure 16), subsequently became part of the CS AKTive Portal.

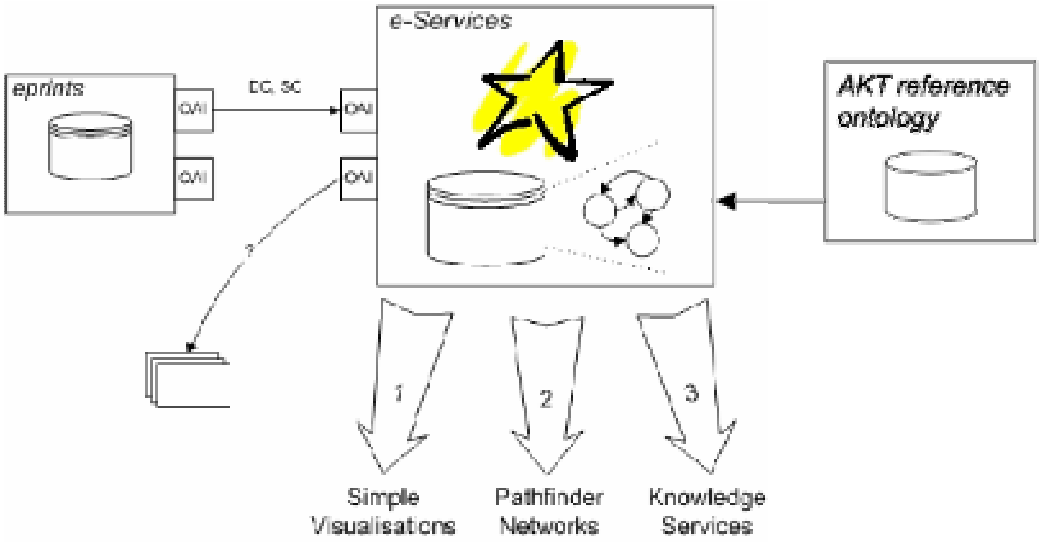


Figure 17: Application of AKT infrastrcture to OpCit

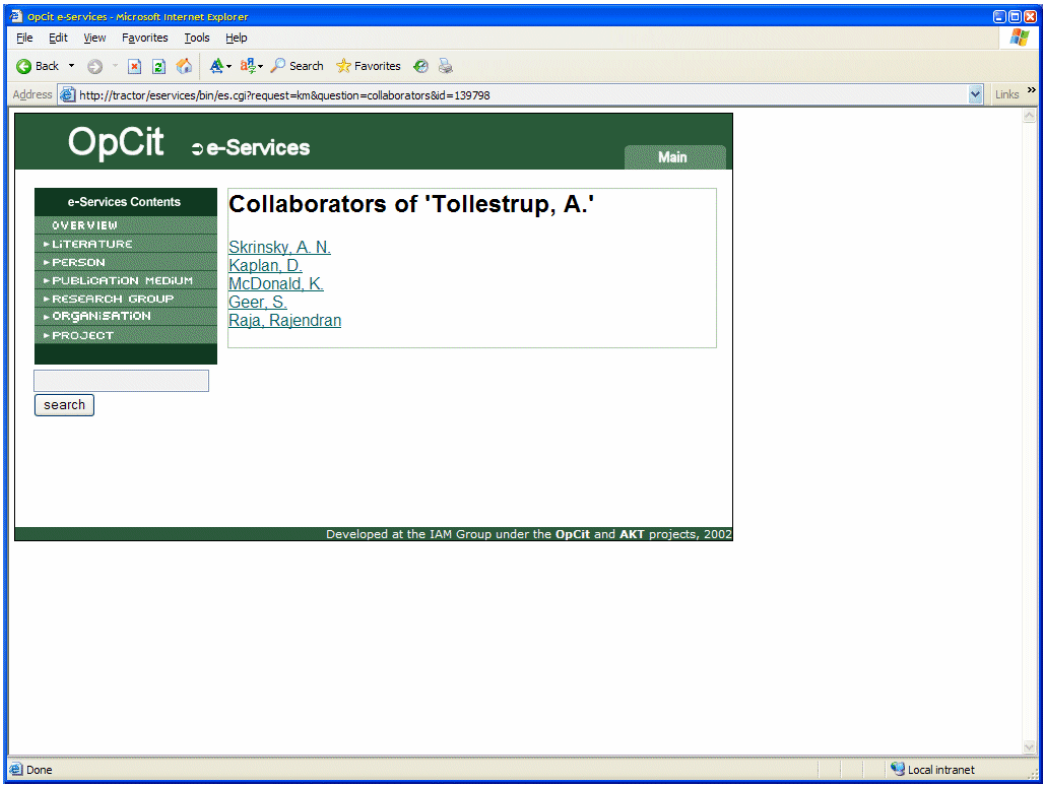


Figure 18: Community analysis

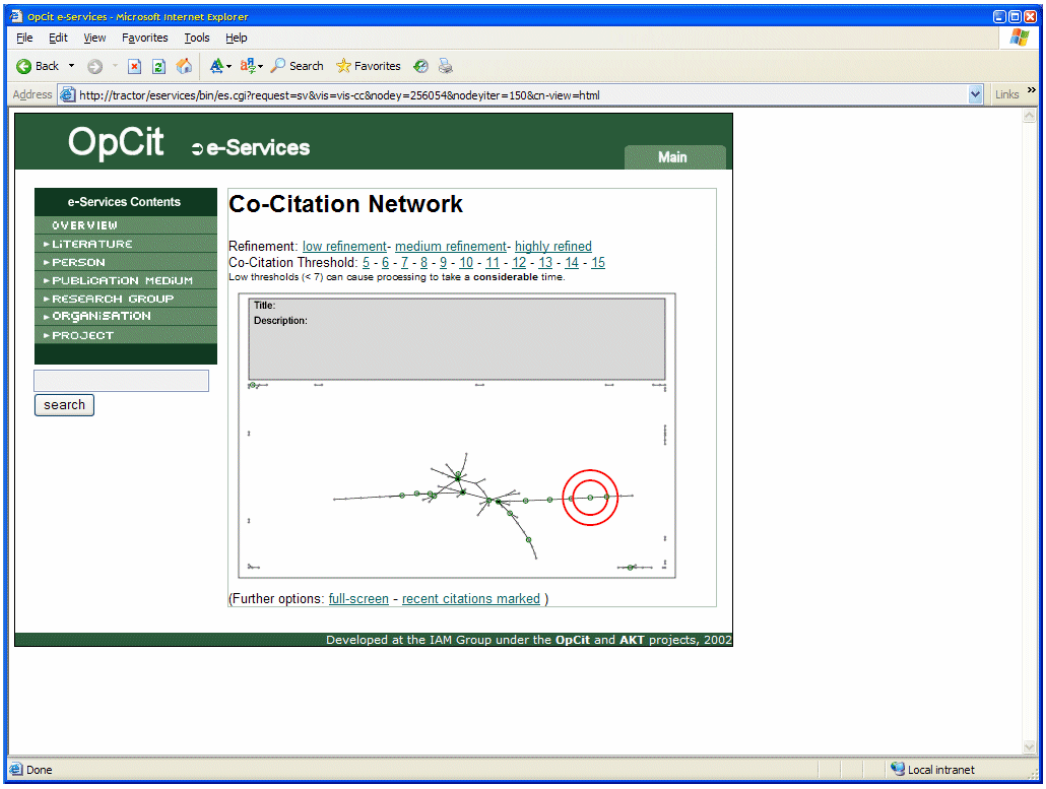


Figure 19: Co-citation network

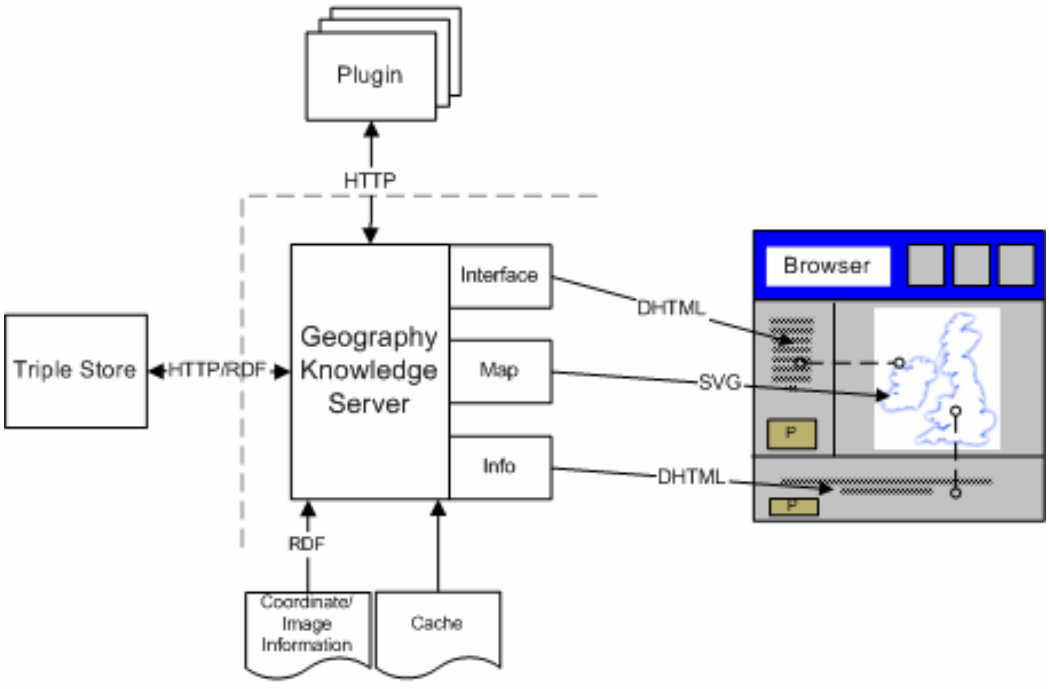


Figure 20: Geographic visualisation

10.2. The Rolls-Royce Test Beds

Two of the main testbeds for AKT in the first half of its funding were provided by Rolls-Royce. Rolls-Royce takes seriously the need to keep accurate records of design,

manufacture and testing; in part this is due to the nature of their business (aerospace) and in part due to the recognition that their main method of capturing corporate memory is through such documentation. As with many organisations, information systems at Rolls-Royce have evolved over time, resulting in a number of document and information management systems. In addition, like many companies in the early 1990s Rolls-Royce outsourced the management of their IT systems. This has greatly influenced the scope of the two testbeds, as it is not possible for AKT to quickly integrate prototype systems for investigating alternative designs, or to carry out software trials. As a result AKT has focussed on using Rolls-Royce data and providing demonstrators (the Intelligent Document Retrieval system and the Designers' Workbench) to show how advanced knowledge technologies can be applied in an advanced international manufacturing company.

10.2.1 Intelligent Document Retrieval (IDR) demonstrator

Within Rolls-Royce Each Operational Business Unit (OBU) is staffed with a number of engineers from different specialisations, as components are usually designed by a number of specialists thereby creating federated (multi-perspective) views. To facilitate easier dissemination of knowledge within a given OBU, each OBU has developed its own, largely hand crafted, website.

In the first phase of this testbed we carried out a number of Knowledge Acquisition interviews with selected engineers, from whom simple ontologies of concepts and relationships were derived. During these interviews the 'card sort' technique (Shadbolt and Burton, 1995) was used to elicit how the engineer referenced different document types, and the relationships between these documents (see Figure 21). A demonstrator was built that showed how, by using a simple ontology, appropriate documents can be retrieved from the document repository. The demonstrator used techniques developed from:

- An ontological hypertext system Conceptual Open Hypermedia Services Environment (COHSE – Carr et al 2001);
- A framework for developing ontologically driven portals (Ontoportal – Kampa et al 2001) which allows different ontologies to be used on the same document set.

Depending on the function (design, stress, thermodynamics or manufacturing) the engineer is undertaking, he/she is presented with appropriate concepts from the ontology. By selecting a concept, appropriate documents are returned; these are ranked based on the document types the engineers most commonly used to undertake the task.

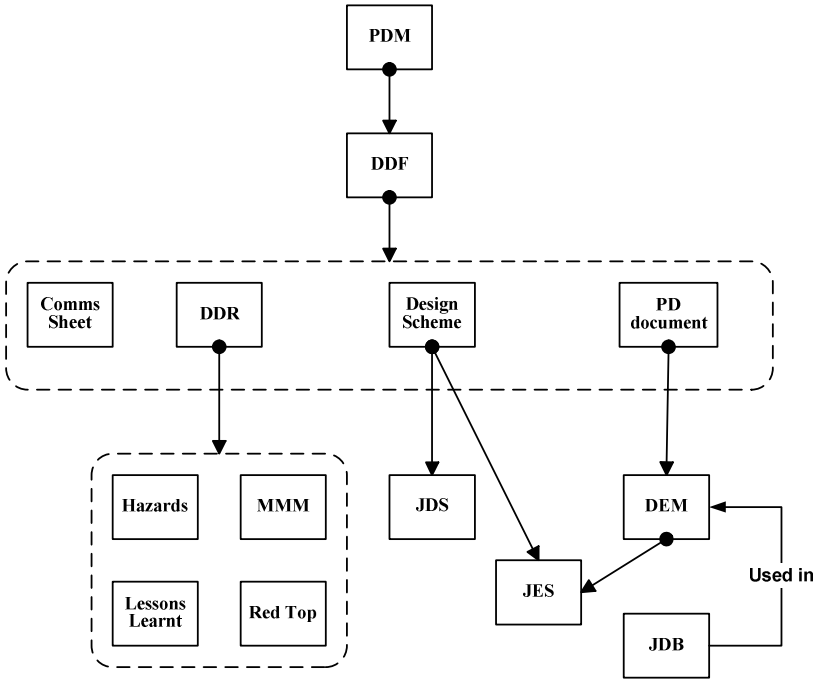


Figure 21: A typical result of a card sort

During this phase, AKT researchers noted that Rolls-Royce were capturing knowledge through hand crafted OBU specific websites, where information can easily be lost (e.g., links removed). Hence, future work will include the development of tools and methods to enable engineers to create semantically enriched meta-data, to facilitate the capture, reuse and maintenance of this knowledge. Additionally, we plan to evaluate the benefits of an Intelligent Document Retrieval system against standard Information Retrieval metrics.

10.2.2 The Designers' Workbench

The primary aim of the Designers' Workbench is to assist designers by checking designs for violations of constraints. These constraints are often in the form of simple, easy to overlook, rules. This will allow the user to focus on more important issues. The Workbench also stores the rationales for the constraints, so that new designers can learn why the rules are used, and so that experienced designers can see if a rule is obsolete, and requires modification.

In the current version of the Workbench, the user can select a feature from an ontology, and use it to annotate an existing drawing. Each type of feature has its own set of properties, and values can be assigned to some or all of the properties of each feature. At any stage, the user can check the constraints. The system will search the features to see which of them (singly or in combination) are affected by constraints. If any constraints are applicable, they are checked, and any violations are reported. Changes to the design or to the constraints can be made so as to remove the violations. The features are represented using RDF, and the search for features affected by a constraint is performed using RDQL. The actual checking of the constraints is done by calls to Sicstus Prolog predicates.

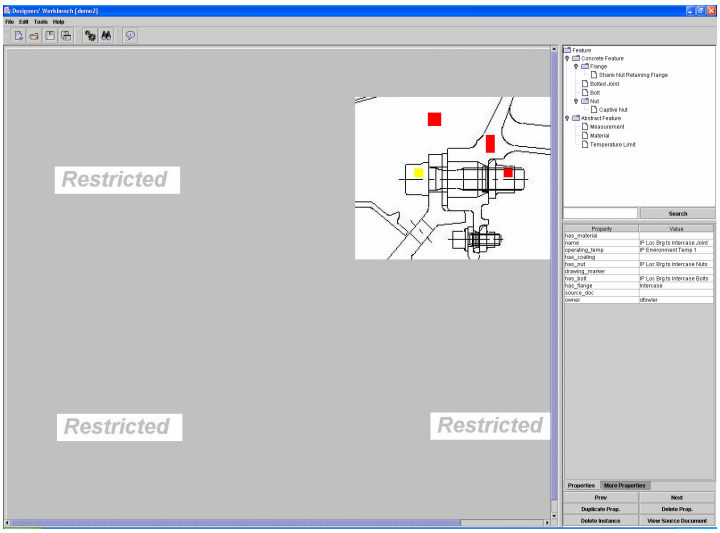


Figure 22: The Designers' Workbench interface

Planned future work on the Workbench includes:

- Integration with existing CAD and knowledge-based engineering systems.
- Recording the decisions that were made during design as unobtrusively as possible.
- Generation of reports from the stored design knowledge, saving work for the designer.
- Intelligent Editors which extract important information from documents semi-automatically
- Case based retrieval of previous, similar, designs
- Tools to allow engineers to capture and maintain their own knowledge bases.

10.3. MIAKT: Supporting Medical Decision Making

Clinical diagnosis and treatment of disease are undergoing a transformation. Clinicians are increasingly required to integrate images and signals of different types, at spatial scales that range from microns (e.g. cells) to mm (MRI) to several cm (EEG), and on temporal scales ranging from microseconds to months. As a result, clinicians are experiencing a deluge of data, arising mainly from images and signals. But this deluge poses as many problems as opportunities for the busy clinician. In fact, clinicians need information not data, where information comprises data plus interpretation for clinical relevance.

The project 'MIAKT Grid enabled knowledge services: collaborative problem solving environments in medical informatics' www.aktors.org/miakt is a joint initiative between the AKT IRC, specialising in knowledge technologies for the management and synthesis of appropriate information and knowledge content, and the MIAS IRC, specialising in the intelligent analysis and handling of medical data. This 24 month project was awarded to the AKT IRC as part of the UK e-Science programme and started in the Summer of 2002. The aim of the project is to apply the capabilities of AKT and MIAS to collaborative medical problem solving using knowledge services provided via the e-Science Grid infrastructure.

The initial focus of the project is the Triple Assessment (TA) method in symptomatic focal breast disease. The domain has been chosen because it contains a number of characteristics that make it especially valuable for the application of knowledge technologies and image analysis. These characteristics include:

- Large amounts of complex data, information and knowledge necessary to inform decision making
- Computationally intensive image and signal interpretation problems
- The collaborative and distributed nature of the task
- Little current support for the knowledge based management of content.

The AKT part of the project is focusing on developing the following:

Ontology Services – We have developed a Triple Assessment (TA) ontology covering concepts and processes in the TA process (Wilson et al 2001, Hu et al, 2003). This includes different stakeholder ontologies, ontologies from different imaging methods (in particular, X-ray mammography and MRI) and ontologies from different aspects of the TA process (in particular radiology and histopathology). Mappings between these ontologies are being developed and maintained. This work has made use of substantial amounts of technology and experience derived from the core AKT project.

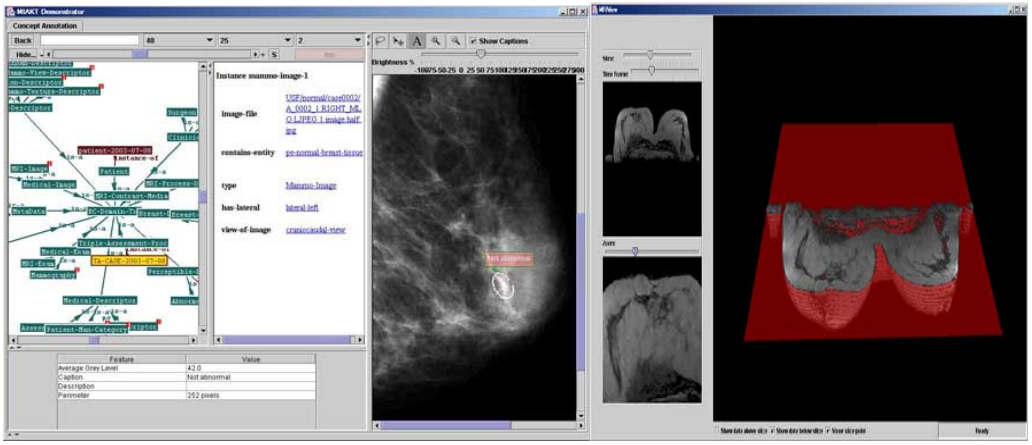


Figure 23 MIAKT Tool: Ontology Annotation with Feature Detection

A set of tools and services are being developed to provide for retrieval of image sets, linking of content, automatic and semi-automatic annotation of content and personalization of content based on individual stakeholder ontologies. We have already experimented with different architectures so that the various modalities of storage, analysis and retrieval can be accommodated (Figure 23). In particular, the 3store developed as part of the core AKT infrastructure is now part of the current architecture. Among other facilities we seek to provide is that of generating natural language summaries of annotated images. Requirements for these tools are being developed with the medical experts as well as image processing specialists from MIAS.

As well as ontology markup the enrichment we are investigating the inclusion of discussion threads, and decision rationales that reflect how the experts have come to particular conclusions.

The other aspects of this project that are principally the responsibility of the MIAS IRC partners includes services invoked over the grid to enact dynamic registration of

images (e.g. aligning images from one patient visit to the next), dynamic construction of fused images. Finally, we are using the IRS architecture (Motta et al 2003, and section 9.3) as a way of integrate web and grid services.

11. The science of semantic web services

AKT, as noted in section 2, is organised around the basic challenges for knowledge management. However, in the complex, distributed world of the Semantic Web, where the problems and contexts will be highly heterogeneous, a similarly heterogeneous set of tools and approaches would be required for complete coverage. Obviously, complete coverage, if not exactly a pipedream, is well beyond the scope even of a large interdisciplinary IRC such as AKT. But AKT, in facing the demands of the likely SW context, has needed to develop many heterogeneous services and technologies, integrated by a unified approach to KM, a single understanding of the likely course of development of the SW, common infrastructure assumptions, and experience of real-world problems as exemplified by the testbeds.

Hence there are various themes that can be detected running through AKT through all the challenges, and which could easily be seen as further organising principles for the project. As well as obvious issues such as the creation, population, use, reuse and visualisation of ontologies, both to structure knowledge repositories and acquisition efforts, and to act as protocols for communication with other services and technologies, a number of interesting questions have been detectable throughout the range of AKT's research across all the challenges (Figure 24). Indeed, many of these issues have presented themselves as pressing precisely because they turned out to influence much of AKT's research, rather than because they were anticipated in advance.

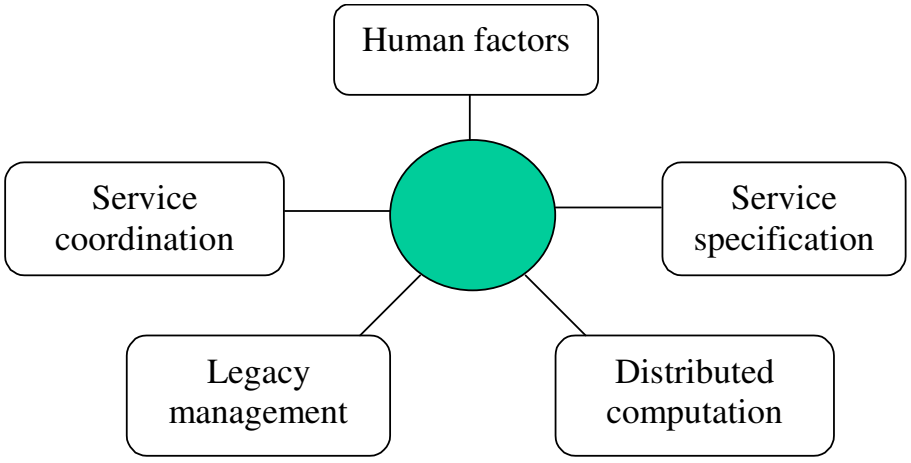


Figure 24: Issues for semantic web services

These issues include:

- Human factors. Technologies and services cannot be the complete solution to any problems, though they will be 50% of a good solution. They must be accepted and appropriate within the organisational context, and for example the production of ontologies will clearly be eased if the ultimate users have the ownership of the development process (Domingue et al 2001).

Indeed, this conception of the importance of the human dimension has always been a key part of the SW (Figure 25). In the well-known diagram of the levels of the SW,

the top level is that of trust. AKT has already been examining theories of trust of both technology and processes, and it is anticipated that this will be a key area for research in the project (O'Hara in press).

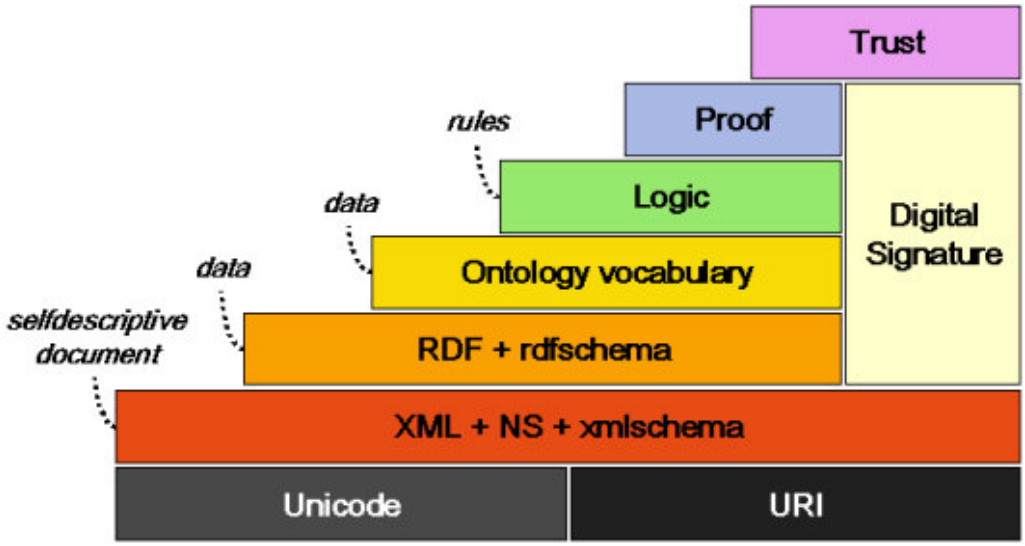


Figure 25: The levels of the Semantic Web

- Service coordination. It is likely that different services will be available from many heterogeneous sources, and efficient use of them will depend on the ability to broker services, and to understand how best to compose services to produce the desired outcome. AKT has addressed these issues under many headings, and answers to these questions would add greatly to the power of the AKT approach (cf sections 4.1, 4.2, 5.2, 6.2.2, 9.2 etc).

- Service specification. Similarly, finding languages for specifying services, coordinating ontologies for sharing understanding, will be a further important set of research issues. We are actively engaged in work to define DAML-S which aims to become a widely adoped Semantic Web Services Language.

- Legacy management. The WWW contains a great deal of material, much of which remains useful for a hard-to-specify quantity of time. Managing this material, gaining the maximal amount of leverage from it, will be crucial for optimising the value of the SW. Put more negatively, it is also essential that the smooth functioning of the SW is not gummed up by the amount of dross that remains on the WWW. In particular, issues to be dealt with include the use of out-of-date formalisms, and the large amount of knowledge – often highly useful knowledge – that is buried in natural language. Clearly AKT has been deeply involved with these issues, and will continue to be so (cf sections 3.2, 3.3, 6.1, 6.2, 8, 9.4 etc).

- Distributed computation. Include problems of scale and inferential precision/completeness

These issues will form part of the research agenda for anyone working to build the next generation of intelligent web services. We anticipate that they will be important forcing functions on our own work and they figure in the explicit work we anticipate for the next phase of AKT.

12. Future work

What we have learnt in the course of the first three years is just how far our objectives require us to confront very significant research challenges that lie at the heart of computer science.

These include (i) the construction and maintenance of multiple ontologies, (ii) at what grain size to construct effective ontologies and how task specific or neutral to make them (iii) the continued challenge of capturing and annotating content on a web scale, (iv) determining the provenance of and establishing trust in knowledge content and annotations on the web (v) managing and dealing with meaning equivalence on the web (when do two descriptions refer to the same object or two sentences mean the same thing), (vi) constructing inference services that are robust in the face of inconsistent and incomplete information, (vii) developing a rich computational notion of context.

An issue that we have focused on more than anticipated in the first phase and that will continue to be important is active task and collaboration support in making use of knowledge in support of organisational and individual objectives.

Undoubtedly we will continue to develop our research so that it can both be exploited and informed by grid based computing. Another general extension of our work will be to ambient intelligence domains where computing services become more pervasive and ubiquitous and the boundary between the physical and digital continues to blur.

We have already mentioned our involvement in the UK OST Foresight Programmes in Cognitive Systems and Cybertrust. In the first of these we envisage developing a stream of work to address the Memories for Life Grand Challenge. In the case of Cybertrust we expect this to become a topic of real urgency as individuals and organisations realise how important and hard it is to build effective computational models of provenance and trust (O'Hara in press, Chapter 5).

13. Conclusion

The original aim of AKT was to provide "joined up" forms of knowledge technologies appropriate to the (then anticipated, now real) challenges of managing knowledge in the large-scale distributed setting afforded by the internet, and to develop the theories underpinning these new technologies. We classified our research into six themes corresponding to typical traditional stages in the lifecycles of such systems (acquisition, modelling, re-use, retrieval, publishing and maintenance) – the idea being to innovate in each of these stages through interactions between them.

Sections 3 to 8 of this document describe progress according to this classification. Our work on acquisition has focused on the harvesting of ontologies from unstructured or semi-structured sources; the key contributions being in the improvement of accuracy in harvesters by exploiting the benefits of large scale and heterogeneity in corpora (these traditionally being viewed as a problem rather than an opportunity) and assessing the leverage gained by provision of minimal ontological backbones. Rather than considering narrowly the modelling of problem solvers or knowledge bases, our modelling efforts tackle issues across the lifecycle – the most obvious of these being to model lifecycles themselves and the coordination between Web services, but we have also gained insights into essential modelling processes such as the mapping and merging of ontologies. Related to this is our research into reuse of Web services via brokering systems and our experiments in mediating between problem solvers via

partially shared ontologies. As the Semantic Web evolves from the traditional Web we are devising retrieval mechanisms that scale to the task of annotating large volumes of semi-structured legacy material and that use such annotations to provide flexible query answering and semantic browsing of Web sites. While our retrieval work has concentrated on the transition from informal to formal media, our work on knowledge publishing demonstrates how formally expressed knowledge may be made more personal by applying human-centred means of interpretation – for example layered querying, template-based personalisation or natural language generation. Our maintenance tools also respond to the human problems associated with inevitable "ontology drift" as use of language in an organisation changes over time – our coreference, mapping and merging systems providing assistance in managing this drift.

Although our themed classification above has provided hooks upon which to hang specific contributions of our research it is not of itself our driving force for innovation. That is provided by the need to provide large scale support of knowledge management in distributed environments. This force has increased considerably since the outset of AKT, thanks to the Semantic Web along with related computation grid and ambient intelligence initiatives. A beneficial consequence is that we now have natural foci for integration of AKT efforts (for example in supporting the provision of Web services) and we are more likely to understand our research in terms of such foci, and the issues they raise, rather than the more traditional classification with which we began. This need not change the nature of the fundamental science but it does change the way we may think about it and connect it to the broader research community. Specific examples of the hot topics now evolving in this new context are:

- Construction and maintenance of multiple ontologies, since Web services may be designed independently but must interact (with clients and with one another). There will be practical issues of grain size and task specificity in addition to the technical issues of legacy management, mapping and merging with which AKT is familiar.

- Managing the evolution from Web to Semantic Web through annotation on a large scale. Automated annotation is sensitive to the nature of the material being annotated – small, rigidly structured texts being capable of much more precise automated annotation than large, free texts for example. This is a spur for empirical analysis and further development of AKT annotation tools.

- Maintaining semantic coherence of as high a degree as reasonably possible between Web services. Semantic equivalence cannot be guaranteed in this sort of open environment so we need ways of managing equivalence of meaning between services – an issue that is pertinent to a range of existing AKT results, from coreference resolution to coordination protocols. We also need ways of building and coordinating services that are robust in the face of inconsistent or incomplete information.

- Specifying services in ways that allow them to be easily described and coordinated. Emerging standards (such as DAML-S for service specification) provide a point of reference but do not in themselves answer the problem of how Web service specifications are most effectively harnessed for essential tasks such as brokering and they do not explain how services should be composed or coordinated. Here AKT can apply its experience in construction of composite services by integrating components; of coordinating groups of services with intersecting ontologies; and of specifying re-usable coordination protocols.

- Determining the provenance of and establishing trust in knowledge content and annotations on the web. This depends on developing a rich computational notion of context that also is practical in use. Perhaps more than most of the other issues above this cuts across a broad range of existing AKT work, from human-centred research into communities of practice to the formal methods used for propagation of properties in our lifecycle calculus. It also draws on the experience gained from our testbed projects.

To tackle problems like those above it is necessary to have gained momentum in relevant theory and technology; to have put in place substantial infrastructure capable of supporting controlled experimentation; and to have the group coherence necessary to focus on areas where AKT can make a difference. This has required substantial investment from all the AKT participants: re-interpreting traditional theory in a new context; devising and adapting technologies to suit shared research goals; building knowledge bases and infrastructure for empirical experiments; and fostering the human contacts which support a common research culture. It has, however, left us well placed to confront the challenges described above.

14. References

Aiken, A. and Sleeman, D. (2003) *Refiner++: A Knowledge Acquisition and Refinement Tool*, Dept CS, University of Abderdeen, Internal Technical Report.

Alani, H., Dasmahapatra, S., Gibbins, N., Glaser, H., Harris, S., Kalfoglou, Y., O'Hara, K. and Shadbolt, N. (2002) 'Managing Reference: Ensuring Referential Integrity of Ontologies for the Semantic Web' Asunción Gómez-Pérez and V. Richard Benjamins (eds.) *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web* (Springer-Verlag 2002) pp.317-334

Alani, H., Dasmahapatra, S., Shadbolt, N. and O'Hara, K. (2003a) 'ONTOCOPI: Using Ontology-Based Network Analysis to Identify Communities of Practice' *IEEE Intelligent Systems*, March/April 2003, 18-25.

Alani, H., Kim, S., Millard, D. E., Weal, M. J., Hall, W., Lewis P. H. and Shadbolt, N. (2003b) 'Automatic Ontology-Based Knowledge Extraction from Web Documents' *IEEE Intelligent Systems*. January/February, 18(1), pp. 14-21.

Alberdi, E. and Sleeman, D. (1997) 'ReTax: A Step in the Automation of Taxonomic Revision' *Artificial Intelligence* 91, 257-279

Bachler, M.S., Buckingham Shum, S., De Roure, D., Michaelides D. and Page, K. (2003) 'Ontological Mediation of Meeting Structure: Argumentation, Annotation, and Navigation' *1st International Workshop on Hypermedia and the Semantic Web*, ACM Hypertext, Nottingham.

Barwise, J. and Seligman, J. (1997) *Information Flow: The Logic of Distributed Systems* Cambridge: Cambridge University Press.

Benjamins, V. R., Plaza, E., Motta, E., Fensel, D., Studer, R., Wielinga, R., Schreiber, G., Zdrahal, Z. and Decker, S. (1998) 'An intelligent brokering service for knowledge-component reuse on the World-Wide Web,' presented at KAW 1998. Available online from http://ksi.cpsc.ucalgary.ca/KAW/KAW98/benjamins3/

Bontcheva, K. (2001) 'Tailoring the Content of Dynamically Generated Explanations' M. Bauer, P.J. Gmytrasiewicz, J. Vassileva (eds). *User Modelling 2001: 8th*

International Conference, UM2001, Lecture Notes in Artificial Intelligence 2109, Springer Verlag.

Bontcheva, K. (2002) 'Adaptivity, Adaptability, and Reading Behaviour: Some Results from the Evaluation of a Dynamic Hypertext System' *Proc. Second International Conference on Adaptive Hypertext*.

Bontcheva, K., Brewster, C., Ciravegna, F., Cunningham, H., Guthrie, L., Gaizauskas, R. and Wilks, Y. (2001) 'Using HLT for Acquiring, Retrieving and Publishing Knowledge in AKT' *Proc.EACL/ACL Workshop on Human Language Technology and Knowledge Management*, Toulouse, France.

Brewster, B. (2002) 'Techniques for Automated Taxonomy Building: Towards Ontologies for Knowledge Management' *Proceedings CLUK Research Colloquium*, Leeds, UK, 2002

Brewster, C., Ciravegna, F. and Wilks, Y (2001a) 'Knowledge Acquisition for Knowledge Management: Position Paper' *Proceeding of the IJCAI-2001 Workshop on Ontology Learning* held in conjuction with the 17th International Conference on Artificial Intelligence (IJCAI-01), Seattle, August, 2001

Brewster, C., Ciravegna, F. and Wilks, Y (2001b) 'User-Centred Ontology Learning for Knowledge Management' *7th International Workshop on Applications of Natural Language to Information Systems*, Stockholm, June 27-28, 2002, Lecture Notes in Computer Sciences, Springer Verlag.

Brewster, C., Ciravegna, F. and Wilks, Y (2003) 'Background and Foreground Knowledge in Dynamic Ontology Construction: Viewing Text as Knowledge Maintenance' *Proceedings of the Semantic Web Workshop*, SIGIR August 2003.

Buckingham Shum, S., De Roure, D., Eisenstadt, M., Shadbolt, N. and Tate, A. (2002) 'CoAKTinG: Collaborative Advanced Knowledge Technologies in the Grid' *Proceedings of the 2nd workshop on Advanced Collaborative Environments, Eleventh IEEE Int. Symposium on High Performance Distributed Computing (HPDC-11)*, July 24-26, 2002, Edinburgh.

Brown, J. S. and Duguid P. (1996). "The Social Life of Documents." *First Monday* 1(1)

Carr, L., De Roure, D., Davis, H. & Hall, W. Implementing an Open Link Service for the World Wide Web, *World Wide Web Journal*, 1(2), 1998, 61-71.

Carr, L., Bechhofer, S., Goble, C. and Hall, W. (2001) 'Conceptual Linking: Ontology-based Open Hypermedia' *Tenth World Wide Web Conference*, Hong Kong.

Chen-Burger Y-H. and Stader, J. (2003) 'Formal Support for Adaptive Workflow Systems in a Distributed Environment' *Workflow Handbook 2003*, Future Strategies Inc., USA, 2003.

Chen-Burger Y-H., Tate, A. and Robertson, D. (2002) 'Enterprise Modelling: A Declarative Approach for FBPML' *European Conference on Artificial Intelligence, Knowledge Management and Organisational Memories Workshop*, 2002.

Ciravegna, F. (2001a) 'Challenges in Information Extraction from Text for Knowledge Management' *IEEE Intelligent Systems and Their Applications*, 16-6, November, (2001).

Ciravegna, F. (2001b) 'Adaptive Information Extraction from Text by Rule Induction and Generalisation' *Proceedings of 17th International Joint Conference on Artificial Intelligence* (2001).

Ciravegna, F. (2001c) '$(LP)^2$, an Adaptive Algorithm for Information Extraction from Web-related Texts' *Proceedings of the IJCAI-2001 Workshop on Adaptive Text Extraction and Mining* held in conjunction with the 17th International Conference on Artificial Intelligence (IJCAI-01), Seattle, August, 2001

Ciravegna, F., Dingli, A., Guthrie, D. and Wilks, Y. (2003) 'Integrating Information to Bootstrap Information Extraction from Web Sites' *IJCAI03 Workshop on Information Integration*, held in conjunction with the International Conference on Artificial Intelligence (IJCAI-03), Acapulco, August, 2003.

Ciravegna, F., Dingli, A., Iria, J. and Wilks, Y. (2003 – submitted) 'Multi-strategy Definition of Annotation Services in Melita' submitted to International Workshop on Human Language Technology for the Semantic Web and Web Services, held in conjunction with ISWC 2003 International Semantic Web Conference, Sanibel Island, Florida, 20-23 October 2003

Ciravegna, F., Dingli, A., Petrelli, D. and Wilks, Y. (2002) 'User-system cooperation in document annotation based on information extraction' *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management, EKAW02*, 2002.

Ciravegna, F. and Wilks, Y. (2003) 'Designing adaptive information extraction for the semantic web in Amilcare' S. Handschuh and S. Staab, editors, *Annotation for the Semantic Web*, Frontiers in Artifcial Intelligence and Applications. IOS Press, Amsterdam, 2003.

Cunningham, H., Maynard, D., Bontcheva, K. and Tablan, V. (2002) 'GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications' *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*. Philadelphia, July 2002.

De Roure, D.C, Jennings, N.R. and Shadbolt, N.R. (2003a) "The Semantic Grid: A Future e-Science Infrastructure" (pages 437-470) in F.Berman, A.J.G.Hey and G.Fox, Grid Computing: Making The Global Infrastructure a Reality, John Wiley & Sons.

De Roure, D.C, Jennings, N.R., Shadbolt, N.R. and Baker, M.A. (2003b) "The Evolution of the Grid" (pages 65-100)) in F.Berman, A.J.G.Hey and G.Fox, Grid Computing: Making The Global Infrastructure a Reality, John Wiley & Sons.

Domingue, J., Motta, E., Buckingham Shum, S., Vargas-Vera, M., Kalfoglou, Y. and Farnes, N. (2001) 'Supporting ontology-driven document enrichment within communities of practice' *Proceedings 1st International Conference on Knowledge Capture (K-Cap 2001)*, Victoria, BC, Canada.

Dzbor, M., Domingue, J. and Motta, E. (2003) 'Magpie – Towards a Semantic Web Browser' *Proceedings of the 2nd International Semantic Web Conference 2003 (ISWC 2003),* 20-23 October 2003, Sundial Resort, Sanibel Island, Florida, USA

Fensel, D. and Motta, E. (2001) 'Structured Development of Problem Solving Methods' *IEEE Transactions on Knowledge and Data Engineering*, 13(6), pp. 913-932.

Fernández-López, M.; Gómez-Pérez, A. Overview and Analysis of methodologies for building ontologies *Knowledge Engineering Review.* Vol. 17(2). 2002. Pags: 129-156.

Fischer, G., McCall, R., Ostwald, J., Reeves, B. and Shipman, F. (1994). Seeding, evolutionary growth and reseeding: Supporting the incremental development of design environments. *Human Factors in Computing Systems (CHI '94)*, Boston, MA (April 24 - 28), ACM Press. 292-298.

Gibbins, N., Harris, S. and Shadbolt, N. (2003) 'Agent-based Semantic Web Services' *Proceedings of the Twelfth International World Wide Web Conference*, pp 710-717

Goble, C., De Roure, D., Shadbolt, N. and Fernandes, A. (2003) "Knowledge and the Grid". To appear as chapter 23 in The Grid: Second Edition, Morgan Kaufmann.

Goguen, J. and Burstall, R. (1992) 'Institutions: Abstract Model Theory for Specification and Programming' *Journal of the ACM* 39(1):95-146, 1992

Gray, P., Hui, K. and Preece, A. (2001) 'An Expressive Constraint Language for Semantic Web Applications' *E-Business and the Intelligent Web: Papers from the IJCAI-01 Workshop,* pp 46 - 53, 2001. A. Preece & D. O'Leary (eds), AAAI Press.

Gruber, T. R. (1993) 'A Translation approach to Portable Ontology Specifications' *Knowledge Acquisition*, 5(2): p. 199-221.

Handschuh, S., Staab, S. and Ciravegna, F. (2002) 'S-CREAM – Semi-automatic CREAtion of Metadata' *Proc. of the 13th International Conference on Knowledge Engineering and Knowledge Management, EKAW02*, Sigüenza, Spain, (2002).

Harris, S and Gibbins, N. (2003) *3store: Efficient Bulk RDF Storage*. http://eprints.ecs.soton.ac.uk/archive/00007970/

Hitchcock, S., Woukeu, A., Brody, T., Carr, L., Hall, W and Harnad, S. (2002) *Evaluating Citebase, an open access Web-based citation-ranked search and impact discovery service*. Final Report to JISC NSF 5nternational Digital Libraries Research Programme. http://opcit.eprints.org/evaluation/Citebase-evaluation/evaluation-report.html

Hu, B., Dasmahapatra, S and Shadbolt, N. (2003) 'Mammographic Ontology: Experience and Lessons' *International Workshop on Description Logics*, Rome, Italy.

Hui, K., Chalmers, S., Gray, P. and Preece, A. (2003) 'Experience in using RDF in Agent-mediated Knowledge Architectures' *Workshop on Agent-Mediated Knowledge Management, AAAI 2003 Spring Symposium.*

Kalfoglou, Y., H. Alani, K. O'Hara, & N. Shadbolt (2002) 'Initiating Organizational Memories using Ontology Network Analysis' *Knowledge Management and Organizational Memories workshop*, *15th European Conf. Artificial Intelligence*, Lyon, France, pp 79-89.

Kalfoglou, Y. and Schorlemmer, M. (2002) 'Information Flow based ontology mapping' *On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, Lecture Notes in Computer Science 2519, pp. 1132-1151. Springer, 2002

Kalfoglou, Y. and Schorlemmer, M. (2003a) 'Ontology mapping: the state of the art' *The Knowledge Engineering Review 18(2), 2003*

Kalfoglou, Y. and Schorlemmer, M. (2003b) 'IF-Map: An Ontology-Mapping Method based on Information-Flow Theory' *Journal of Data Semantics*, Springer, 2003

Kampa, S. (2002) *Who are the experts? E-Scholars in the Semantic Web.* PhD Thesis. University of Southampton.

Kampa, S., Miles-Board, T and Carr, L. (2001) 'Hypertext in the Semantic Web' *Proceedings ACM Conference on Hypertext and Hypermedia 2001*, pages 237-238, Aarhus, Denmark.

Kim, S., Alani, H., Hall, W., Lewis, P., Millard, D., Shadbolt, N. and Weal, M. (2002) 'Artequakt: Generating Tailored Biographies with Automatically Annotated Fragments from the Web' *Proceedings Semantic Authoring, Annotation and Knowledge Markup Workshop* in the 15th European Conference on Artificial Intelligence, Lyon, France.

Kushmerick, N., Weld, D. and Doorenbos, R. (1997) 'Wrapper induction for information extraction' *Proc. of 15th International Conference on Artificial Intelligence*, Japan (1997).

Lei, Y., Motta, E., Domingue, J. (2003) Design of Customized Web Applications with OntoWeaver. In Proceedings of the 2nd International Conference on Knowledge Capture (K-CAP'03), Sundial Resort, Sanibel Island, FL, USA, October.

Lewis, P., Davis, H.. Dobie, M. & Hall W. (1996). Towards multimedia thesaurus support for media-based navigation. In *Proceedings of the First International Workshop on Image Databases and Multimedia Search*.

Leonard, T. and Glaser, H. (2001) 'Large scale acquisition and maintenance from the web without source access' Handschuh, Siegfried and Dieng-Kuntz, Rose and Staab, Steffan, Eds. *Proceedings Workshop 4, Knowledge Markup and Semantic Annotation, K-CAP 2001*, pages 97-101.

Lieberman, H., Fry, C. and Weitzman, L. (2001) 'Exploring the web with reconnaissance agents' *Communications of the ACM*. 44(8): p. 69-75.

McLuhan, M. (1962) *The Gutenberg Galaxy* London: Routledge & Kegan Paul.

Middleton, S., Alani, H., Shadbolt, N. and De Roure, D. (2002) 'Exploiting Synergy Between Ontologies and Recommender Systems' *The Semantic Web Workshop, World Wide Web Conf.*, (WWW'02), Hawaii, USA, pp. 41-50.

Motta E. (1999). Reusable Components for Knowledge Models. IOS Press, Amsterdam.

Motta, E., Buckingham Shum, S. and Domingue, J. (2000) Ontology-Driven Document Enrichment: Principles, Tools and Applications. *International Journal of Human Computer Studies.* 52(5) pp. 1071-1109.

Motta, E., Domingue, J., Cabral, L. and Gaspari, M. (2003) 'IRS-II: A Framework and Infrastructure for Semantic Web Services' *2nd International Semantic Web Conference (ISWC2003)* 20-23 October 2003, Sundial Resort, Sanibel Island, Florida, USA.

Mulholland, P., Zdrahal, Z., Domingue, J., Hatala, M. and Bernardi, A. (2001) A Methodological Approach to Supporting Organisational Learning. *International Journal of Human Computer Studies.* Vol. 55, No. 3, September 1, 2001, pp. 337-367

Niles, I. and Pease, A. (2001) 'Towards a Standard Upper Ontology' *2nd International Conference on Formal Ontology in Information Systems (FOIS-2001*).

Nordlander, T., Brown, K. and Sleeman, D. (2002) *Identifying Inconsistent CSPs by Relaxation*, AUCS/TR0304, CSD University of Aberdeen.

O'Hara, K. (2002) *Plato and the Internet* Cambridge: Icon Books.

O'Hara, K. (2002b) The Internet: A Tool for Democratic Pluralism? Science as Culture, 11(2), pp. 287-98.

O'Hara, K. (in press) *Trust: From Aristotle to Enron* Cambridge: Icon Books.

O'Hara, K., Hall, W., van Rijsbergen, K. and Shadbolt, N. (2003) 'Memory, Reasoning and Learning' available from the UK Government Foresight Project, Cognitive Systems (http://www.foresight.gov.uk)

O'Hara, K. and Shadbolt, N. (2001) 'Issues for an Ontology for Knowledge Valuation' IJCAI workshop on *E-Business and the Intelligent Web*, Seattle, August 5th, 2001, http://www.csd.abdn.ac.uk/ebiweb/papers/ohara.doc

O'Hara, K., Shadbolt, N. and Sleeman, D. (2000) *A Review of the Psychological Literature on Forgetting* Internal document, University of Southampton.

Potter, S. and Aitken, S. (2001) An Expert System for Evaluating the Knowledge Potential of Databases. In Proceedings of the 21st SGES International Conference on Knowledge Based Systems and Applied Artificial Intelligence (ES2001), Cambridge, UK, December.

Prosser, P. (1994) 'Binary constraint satisfaction problems: Some are harder than others' *Proceedings ECAI-94 (11th European Conference on Artificial Intelligence)*, pp. 95-99.

Robertson, D. and Schorlemmer, M. (2003) *F-Life*. URL: http://www.aktors.org/technologies/F-Life, July 2003.

Schacter, D. L. (2001) *The Seven Sins of Memory* New York: Houghton Mifflin.

Schorlemmer, M. and Kalfoglou, Y. (2003) 'Using Information-Flow Theory to Enable Semantic Interoperability' *Proceedings of the 6th Catalan Conference on Artificial Intelligence (CCIA '03),* Palma de Mallorca, Spain, October 2003. Frontiers of Artificial Intelligence and Applications, IOS Press.

Schorlemmer, M. and Kalfoglou, Y. (2003 – submitted) 'On Semantic Interoperability and the Flow of Information' *Proceedings of the ISWC '03 Workshop on Semantic Integration,* Sanibel Island, FL, October 2003.

Schorlemmer, M., Potter, S. and Robertson, D. (2002a) *Automated Support for Composition of Transformational Components in Knowledge Engineering*. Technical Report EDI-INF-RR-0137, Division of Informatics, The University of Edinburgh, June 2002.

Schorlemmer, M., Potter, S., Robertson, D. and Sleeman, D. (2002b) *Knowledge Life-Cycle Management over a Distributed Architecture.* Expert Update 5(3):2-19, 2002.

schraefel, m. c., Karam, M. and Zhao, S. (2003) 'mSpace: interaction design for user-determined, adaptable domain exploration in hypermedia' *Proceedings of AH2003 Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems*, Budapest.

Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., Van de Velde, W. and Wielinga, B. (2000) *Knowledge Engineering and Management*. MIT Press.

Shadbolt, N.R. and Burton, M. (1990) 'Knowledge elicitation' J.R. Wilson & E.N. Corlett, Eds., *Evaluation of Human Work: A Practical Ergonomics Methodology*, pp.321-345. London: Taylor and Francis.

Shadbolt, N.R. and Burton, M. (1995) Knowledge elicitation: a systematic approach, in *Evaluation of human work: A practical ergonomics methodology 2nd Edition* J. R. Wilson and E. N. Corlett Eds, Taylor and Francis, London, England, 1995. pp.406-440. ISBN-07484-0084-2.

Shadbolt, N. R. (2001a) Caught up in the Web in *IEEE Intelligent Systems*, May-June 2001.

Shadbolt, N. R. (2001b) Switching on to the Grid in *IEEE Intelligent Systems*, Jul-Aug 2001.

Shadbolt, N., schraefel, m. c., Gibbins, N. and Harris, S. (2003) 'CS AKTive Space: or How We Stopped Worrying and Learned to Love the Semantic Web' ECS Technical Report, University of Southampton.

Shadbolt, N. (ed.) (2003b) Advanced Knowledge Technologies: Selected Papers, ISBN 0854-327932.

Sleeman, D., Robertson, D., Potter, S. and Schorlemmer, M. (2003) 'Ontology Extraction for Distributed Environments, Knowledge Transformation for the Semantic Web' *Frontiers in Artificial Intelligence and Applications* 95, pp. 80-91. IOS Press.

Tate, A (2003) '<I-N-C-A>: an Ontology for Mixed-initiative Synthesis Tasks' *Proceedings of the Workshop on Mixed-Initiative Intelligent Systems (MIIS)* at the International Joint Conference on Artificial Intelligence (IJCAI-03), Acapulco, Mexico, August 2003.

Uschold M. and Gruninger M. (1996). Ontologies: Principles, Methods and Applications. *Knowledge Engineering Review*, 11(2), pp.93-136.

Vargas-Vera, M. (2001) Invited Speaker at XXXIV National Congress of Mathematics, Toluca, Mexico, 8 October.

Vargas-Vera, M., Motta, E., Domingue, J., Lanzoni, M., Stutt, A. and Ciravegna , F. (2002) 'MnM: Ontology driven semi-automatic or automatic support for semantic markup' *Proc. of the 13th International Conference on Knowledge Engineering and Knowledge Management, EKAW02*, Sigüenza, Spain (2002).

Vargas-Vera, M., Motta, E. and Domingue, J. (in press) 'An Ontology-Driven Question Answering System (AQUA)' to appear in: *New Directions in Question Answering*, MIT Press.

Vasconcelos, W., Robertson, D., Agusti, J., Sierra, C., Wooldridge, M., Parsons, S., Walton, C. and Sabater, J. (2002) 'A Lifecycle For Models of Large Multi-agent Systems' *Proc. of the 2nd International Workshop on Agent Oriented Software Engnering*, Montreal Canada. Springer-Verlag Lecture Notes in Computer Science 2222.

Walsh, T. (2001) 'Search on high degree graphs' *IJCAI-2001*, pp. 266-274.

Walton, C. and Robertson, D. (2002) *Flexible Multi-agent Protocols*. Technical report EDI-INF-RR-0164, Informatics, University of Edinburgh.

Wenger, E. (1998) *Communities of Practice: Learning, Meaning and Identity*. Cambridge University Press, Cambridge.

White, S. and Sleeman, D. (2000) 'A Constraint-Based Approach to the Description & Detection of Fitness-for-Purpose' *ETAI*, vol. 4, pp. 155-183.

Wilson, R., Asbury, D., Cooke, J., Michell, M. and Patnick, J. (2001, Eds.) *Clinical Guidelines for Breast Cancer Screening Assessment*, NHSBSP Publication No 49 April 2001, available at http://www.cancerscreening.nhs.uk/breastscreen/publications/nhsbsp49.pdf

Winter, M. and Sleeman, D. (1995) 'Refiner+: An Efficient System for Detecting and Removing Inconsistencies in Example Sets' *Research & Development in Expert Systems XII*, 115-132.

Winter, M., Sleeman, D. and Parsons, T. (1998) 'Inventory Management using constraint satisfaction and knowledge refinement techniques' *Knowledge Based Systems*, 11, pp 293-300.