

CS AKTive Space: or How We Stopped Worrying and Learned to Love the Semantic Web

Nigel R. Shadbolt, monica m.c. schraefel, Nicholas Gibbins, Stephen Harris

Department of Electronics and Computer Science
University of Southampton
Southampton, United Kingdom
{nrs, mc, nmg, swh}@ecs.soton.ac.uk

Abstract. We present a Semantic Web application that we call CS AKTive Space. The application exploits a wide range of semantically heterogeneous and distributed content relating to Computer Science research in the UK. This content is gathered on a continuous basis using a variety of methods including harvesting and scraping as well as adopting a range models for content acquisition. The content currently comprises around five million RDF triples and we have developed storage, retrieval and maintenance methods to support its management. The content is mediated through an ontology constructed for the application domain and incorporates components from other published ontologies. CS AKTive Space supports the exploration of patterns and implications inherent in the content and exploits a variety of visualisations and multi dimensional representations. Knowledge services supported in the application include investigating communities of practice: who is working, researching or publishing with whom. This work illustrates a number of substantial challenges for the Semantic Web. These include problems of referential integrity, tractable inference and interaction support. We review our approaches to these issues and discuss relevant related work. Socio technical issues are outlined that are seen to be critical for the success or failure of our endeavour.

1. Introduction

Within the Semantic Web community, the field has reached a point where we wish to move from either language definition or the development of individual Semantic Web services towards a test of the larger Semantic Web project. We need to demonstrate the integration of both dynamic content acquisition/delivery and support services. There are numerous aspects to pulling together such a demonstrator application: how the content will be harvested and stored; there are referential integrity problems to address; queries over the space must be time efficient in returning results, and be robust in order to scale; services must communicate with each other and the content; the interaction and visualization must afford both an effective model of the aggregated content as well as a means to interrogate the content meaningfully.

CS AKTive Space (CAS) represents our effort to consider these issues directly in the context of an integrated semantic web application within the AKT framework (www.aktors.org). The mission of AKT is to support the complete knowledge cycle. The complete knowledge cycle runs from acquisition and capture of knowledge

content, through its modelling, retrieval, reuse, publication and maintenance. CS AKTive Space attempts to embrace all elements of the life cycle as well as embodying the interface affordances required for users to engage with this content in a productive way. 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 as well as other models for content acquisition. The content currently comprises around five million RDF triples and we have developed storage, retrieval and maintenance methods to support its management. The content is mediated through an ontology [2] constructed for the application domain and incorporates components from other published ontologies [24].

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 [29] Knowledge services supported in the application include investigating communities of practice [9] and scholarly impact. 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.

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 content presents both threats and opportunities in terms of the sorts of inferential services we can support; how we present the content so that inherent patterns and trends can be directly discerned must be considered; how all this information is to be maintained and sustained as a community exercise is a critical discussion which needs to take place to enable the work to proceed.

In this paper, we present the motivating scenario for CS AKTive space that has provided a context for integration. We next describe the components of CS AKTive Space and the processes that can be run as a result. We describe the interaction design approach we use to integrate these components at the user interface (UI) level. We review related work and discuss the particular research challenges presented by the scenario, and our results to date. Finally, we discuss how this work is to be progressed and the various social and technical challenges that lie ahead.

2 Motivating Scenario for CS AKTive Space

Scenario design is a typical method in both software engineering and human computer interaction to explore development paths and to test the viability of an

approach, from the general to the specific levels of a system [6].

In the case of CS AKTive Space, we developed a scenario both to explore the ways in which we could integrate and present the services we have been developing across the AKT project, as well as to have a litmus test against which we could validate the groundedness of our efforts.

To this end, we initiated a motivating scenario that included a particular user, interested in investigating the CS domain. This scenario was then further refined and validated through design review sessions with the potential scenario stakeholders, in this case, members of the CS Research community and executive of a UK granting organization. The following section presents an overview of that scenario. The italicized terms refer to the services the AKT partners have developed and which we are integrating for the scenario. The italicized phrases indicate the technologies being used. The corresponding numbers are associated with the service names, descriptions and references.

Research Scenario A director of a granting organization is on the road, but has taken work from the office with her: one project she has is to get a workshop happening to consider research issues for Artificial Intelligence and HCI in the UK. The director wishes to know who the best people are in their respective fields in the UK to speak to these ideas at a workshop. She goes to CS AKTive Space (CAS) site and *selects the topics of interest* and the number of names she wishes returned (1). This returns a list of the top 5 established people in each field. She *gathers* (2) the names of the potentials in order to email them later. She then wonders who the *up and comers may be* (1) in the area, and adds these to her list for contact. After spending some more time with the CAS to determine these factors, the director observes some unexpected *correlations among region, area and funding* (3) and becomes interested in encouraging collaborations in the North. She thinks the event should take place around Manchester and therefore wants to do a first pass at *determining venues*(4) that are both enticing and good value for money. She also does a quick check with the *Conflict Calendar*(5) to see when the speakers mayn't have other conference or school-based obligations for a possible workshop date. These dates *get added to her collection of gathered information from the current CAS session* (2). The director goes through the list, and *creates a To Do/Action Items list* (5) based on that information, and *assigns people to the actionable items* (5). With this information in hand, she *contacts* (7) two of her colleagues to go over a few of the details and to confirm task delegation. While going through the information, one of her colleagues, now also looking at the CAS, looks at the *profile page* (6) of one of the candidate speakers, points in passing to *an article about one of the projects* (7) by one of their targeted researchers, noting the commercial transfer that seems to be taking place. Another colleague looks at the paper listing for the researcher (8) and notices a name he recognizes from outside CS seems to be a regular collaborator. He *pulls in additional data* (9) about this person's work. Perhaps they will invite a few relevant non-CS people, and perhaps this researcher should be one? The new name gets moved to the *Action Item list* for followup (3) The director, looking at another researcher's *COP listing* (10) notices that there is already some strong collaborative ties between this researcher in AI and another in their HCI list. Everyone is very pleased by this, and they continue to make plans.

Services Referenced (1) The 3Store [1] supports the queries on the CS Domain, leveraging data from multiple sources which reflects publications to grants to community of practice. Within the CAS User Interface (UI), query information can be filtered to include or not various attributes captured in the data. In the case of up and comers, one might wish to see the list generated with Grants held as PI and Significant Awards turned off, but with Significant Papers and Community of Practice left on. (2) is Hunter Gatherer [31] is used for collecting data from within Web pages and having that data automatically added to a new Web page representing that collection. (3) The Geography Visualizer, a new service recently developed for this project, supports 2 effects: first query results' geographical elements are visualized on the map; second, queries can be constrained by selecting attributes on the map. This is described further in the Information Affordance section, below. (4) Granite Nights [3] is a dynamic filtered query style [8] service that helps find hotels and associated information such as likely restaurants and entertainment venues for a visit to a location. (5) Meet-o-Matic [5] is the basis for Conflict Calendar, (CC) which is currently being adapted to look at event schedules to help determine likely times for meetings. In this case the CC considers publicly scheduled conferences and university schedules based on area and school of relevant participants. (6) IX panel supports the generation of action items/tasks for multiple users. In this case, the IX panel's information is based on both the context of the current CAS session and the information given it specifically by the director from her Hunter Gatherer session. (7) IX panel communicates directly with Buddy Space [18] , a Jabber-based synchronous communication service. (8) The Profile Page is based on current AKT work to generate web pages dynamically from detail in a triple store. (8) The Profile Page is supplemented by the MyPlanet [17] service that associates news stories with people and projects listed in the ProfilePage. (9) Armadillo [15] is a service for on-the-fly, user-determined directed knowledge acquisition from web pages, which can be used to opportunistically expand the knowledge base (10) The Community of Practice (COP) service[10] presents the list of researchers, in this case within the UK only who have collaborated either on papers or projects with the selected researcher. While the CAS UI primarily represents the CoP as a list, the CoP tool, available from the CAS UI, affords other network visualizations of the CoP, affording additional perspectives on the data.

The above scenario captures 10 services afforded on the data, presented within an integrated user interaction experience to support and *the context of use throughout the uses' task cycle*. In the past four months since generating and testing the scenario, we have integrated half these AKT- developed services; we are working on integrating the remainder,

The following sections outline the challenges in the back end and the front end that have been addressed to realize the scenario. We then contextualize this research in terms of related work. The conclusion addresses the challenges for integrating the remaining features in the scenario, and situates our findings to date within the Semantic Web community

3.1 Pragmatic Exploitation of Distributed Content

The scenario that forms the motivation for the CS AKTive Space presupposes that a range of content is available for use by the system. As it stands, some of this content already exists in suitable structured forms, while others do not. We adopt a pragmatic attitude that reflects that fact that although the content that we are gathering is the prime mover that drives the interface, we should also be tolerant of inconsistencies in that content. We use a relatively scruffy approach in which we make the immediate best use of the available data sources, perhaps in an imperfect fashion, while anticipating that we will be able to make better use of them in future. Although this comes at a cost (there is an implicit commitment to future knowledge maintenance), such early exploitation of available content is necessary to initiate a community process that should be self-sustaining in the future and so justify the investment of effort.

3.2 Ontology-mediated knowledge acquisition

Even in a distributed environment like the Semantic Web, the physical distribution of data is much less important than the semantic distribution of data brought about by the use of disparate ontologies for the same application domain. However, we consider ontology translation to be beyond the scope of this study, and so commit to using a single common ontology to express the data which drives the AKTive Space. We use this common ontology, the AKT Reference Ontology [2], to mediate and guide the integration of the different data sources. When expressed in terms of our ontology, the RDF data obtained from these sources are made publicly available through the Hyphenhyphen.info web site (Hyphen being the name for the knowledge acquisition effort) [4] and asserted into an RDF triple store repository which provides the necessary query and inferential capabilities on which the AKTive Space is built.

Due to the wide variety of the data sources that we use, we have found it necessary to invest a degree of effort in developing individual mediators for each of our data sources that recast them in terms of our ontology. These mediators range from specialized database export scripts to XML transformation tools that have been trained to extract the required content from semi-structured web pages [22]. Although these mediators are based on a common framework of code (which handles the rote work of database access, HTTP retrieval, RDF construction and the more common patterns in our ontology, such as date and time expression), they each contain specialized capabilities that are tailored to the content and nature of the individual data sources. While the bulk translation of instance data by such a mediator is straightforward, we believe that the mapping of existing structured and semi-

structured data at the schema/ontology level is not a task that can be effectively automated; the investment of effort in building mediators for our common ontology is reflected in the consequent perceived value of the knowledge base to which they contribute.

3.3 Push and pull models for knowledge acquisition

We employ both push and pull models of knowledge acquisition, where push and pull refer primarily to whether the publisher or consumer are responsible for translating the data into a form which is suitable for the consumer. The push model involves a data source (the publisher) choosing to express its data in terms of the ontology used by the CS AKTive Space. The publisher is solely responsible for the translation, so the consumer may simply retrieve the translated knowledge base without any further effort required on its part. In comparison, the pull model requires that the consumer takes a raw data source (which may be published against some other ontology, or which may only exist as a set of unannotated webpages) and construct a knowledge base from that data source.

In some ways, the pull model has advantages over the push model, in that the consumer has a much greater level of control over what information is encoded within the resulting knowledge bases and is better placed to be able to correct inconsistencies or to adapt to changes in the underlying common ontology, but this comes at the expense of a greater cost to the consumer, both in the acquisition phase of the knowledge lifecycle (when a new data source is acquired), but particularly in the maintenance phase.

We use the pull model predominantly for large, comparatively static data sources (for example, the list of countries and administrative regions given by ISO3166), and as an interim solution for high-value data sources that are of general interest to the community (for example, the Engineering and Physical Sciences Research Council's database of research funding) as a means to 'pump-prime' the system with sufficient data to encourage other members of the community to participate by offering to push their local data sources to us (the viral, rich-get-richer phenomenon that we later describe needs to start somewhere). In the longer term, we aim to encourage the owners of the majority of these pull model sources to move to a push model of delivery.

3.4 Size Matters

The data used in this system is the hyphen.info RDF data describing the academic computer science community in the UK against the OWL/RDF version of the AKT Reference Ontology. The hyphen metadata is 430MB of RDF/XML files containing around 5 million RDF triples describing 800,000 instances of people, places, publications and other items of interest to the academic community.

The ability to store and query this rapidly changing data lead to the requirement for an RDF triple store capable of storing RDF data on the order of 10 million triples (to allow for planned expansion), and updating around 10% of that daily, in a reasonable

time.

To meet these requirements the RDF data is stored in a 3store server. 3store is a Free Software (GNU General Public Licence) RDF triple store that stores RDF triples in a SQL database, using an ontology independent schema for flexibility, scalability and efficiency reasons. The architecture is similar to the ones used in HYWIBAS [25] and SOPHIA [7], though with a greater emphasis on query execution speed and with an underlying RDF representation replacing the frame logic representation.

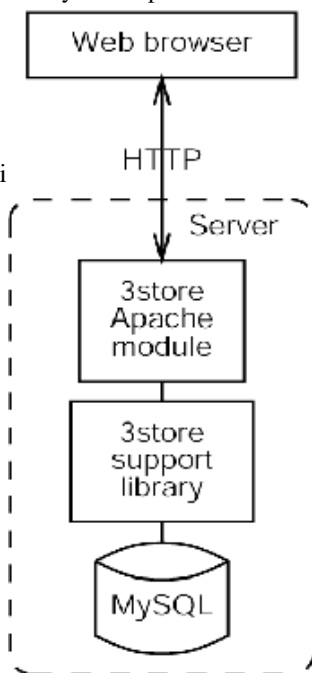
The HYWIBAS design is based around a three level architecture, with the top level performing mappings from the query language (in this case RDQL) into the knowledge format, the middle layer handling the representation of types and classes and the bottom layer, being the RDBMS, storing the fundamental relations. This design allows the system to perform query optimisations at each level of abstraction and the final query is translated into one SQL query and so can be executed by the database engine, in a conventional RDBMS manner, rather than as fragmented queries.

The top level is an apache server module, the middle level is a C library that forms the core of 3store, and the bottom level is provided by the Open Source RDBMS MySQL specifically, rather than being general SQL or using a mapping layer. This allows 3store to take advantage of MySQL strengths, and to use MySQL specific extensions where applicable.

In order to perform efficient inference it is necessary to address the trade-off between query-time complexity and storage complexity [33], consequently we adopt a hybrid eager / lazy system in which some entailments are generated at query time and others are performed at assertion time (when the ontology is added to the triple store).

This design brings the execution time of typical RDQL queries used by the interface down to a few milliseconds, and allows for RDF(S) data files to be asserted at a rate of around 1000 triples/second on a commodity x86 based server, even with large knowledge bases. This allows the interface to be queried and the KB to be updated interactively.

Illustration 1, right: Communications architecture of the CAS system



This system could equally well have been implemented with a relational database as a back-end, such as the one described in [Oliveira and Medeiros, 2000], indeed the low level data storage is being provided by a conventional RDBMS, however the ontological schema used to query and express this data can allow more efficient and effective integration of heterogeneous data sources [27].

3.5 Referential Integrity in the Semantic Web

The current development of knowledge services on the Semantic Web raises a number of issues which are not commonly encountered in existing knowledge based systems, and which pertain to the distribution of knowledge and the difficulty of obtaining agreement on a conceptualisation in a distributed environment when there is no ultimate authority. One such issue is that of coreference, which arises when more than one Uniform Resource Identifier [11] is used to refer to a given resource, and which causes particular problems when statements from different knowledge bases are to be combined.

While the semantics of the URIs used as names by the Semantic Web require that identical names must refer to the same entity, they do not require the unique names assumption. In the absence of any coordination between data sources, it is reasonable to assume that in most cases the unique names assumption holds within the domain of a single data source, but that it does not hold across the union of several data sources. Two data sources on the Semantic Web may contain statements that refer to the same resource, but by different URIs. When constructing a system like the CS AKTive Space which uses knowledge from different sources, we must be able to integrate the contents of these sources by mapping the URIs used by one source onto the URIs used by another.

In addition, Semantic Web resources are not necessarily digital artifacts like web pages which can be retrieved by resolving the URI which names that resource [23]; URIs can also denote resources which are physical entities, such as people or organizations. Since URIs need not denote digital objects, so it is not possible to resolve them and determine if two URIs are coreferent by comparing the objects that they denote. In fact, it is current common practice for Semantic Web agents to treat URIs as opaque symbols and to ignore any retrieval semantics that might otherwise be associated with them (for example, the structure of an HTTP URI gives the necessary means for an agent to be able to retrieve the object indicated by that URI). Unidentified coreferent entities present a problem for Semantic Web applications since they partition the information space in such a way as to reduce the recall of queries made in that space. Even if a SW application is able to use heterogeneous data sources by mediating them through a common ontology, it cannot be properly considered to have integrated those data sources if unnecessary coreferences exist for a resource that is described in more than one data source.

Within the context of a community resource like the CS AKTive Space, the most appropriate - and the simplest - solution to this issue is predominantly social in nature. The emerging knowledge base that is being collaboratively built by the community may be viewed as a gazetteer or name authority containing the agreed names that should be used to refer to entities in the application domain. It is here that the careful selection of data sources in the initial knowledge acquisition phase is important, since the resources that they describe are likely to be referred to in many other sources.

Where the social process that drives this constructive approach to coreference management would prove too expensive, it is complemented by the use of heuristic techniques for identifying coreferent entities [9] and coalescing them in a semi-automatic manner.

4 Information Affordances

(Illustration 2. CAS UI prototype mock up featuring multicolumn queries and geographical visualizer and constraint controller)

4.1 Overview of CAS Interaction Design

In the following section, we present an overview of the CAS interaction model. We follow this with a specific example of how the interaction events communicate with the 3store.

Domain Exploration For the interaction design of CAS, we took a different approach from other Semantic Web systems, as described below in Related Work. In systems like K2A and Ontoport, the interaction model is a browser. With Web browsers where the main interaction is through a click, selections are revealed one element at a time. Users are likewise constrained to evaluate results one link at a time. This iterative approach both to building a query and assessing the result means that users are limited in the efficacy with which they can build queries, and the ways they can engage the results. With respect to the CAS scenario, we needed to make richer kinds of queries and manipulations available to users. To this end, the interaction design reflects the following design goals:

- to afford simple fast queries within the CS domain, such as “what is Nigel Shadbolt’s email address?”

- to support richer kinds of queries like “with whom does Shadbolt collaborate?” “which areas of the country are receiving most funding and in what areas?”
- to support dynamic views of the information to support users’ interests –
- to afford multiple perspectives from which to orient the information space.

To achieve these goals, we focused on *Domain Exploration* rather than generic browsing. That is, the interaction design affords an overview of the domain itself – in this case, the domain is CS research in the UK – based on the ontology of the domain. Based on users’ focus in one area of the domain, the rest of the represented areas reflect that focus [29, 30]. For instance, if a person selects a topic or institution or person or project, the rest of the domain reflects that selection. If Artificial Intelligence is the selected topic, the multipaned window will populate one pane with universities doing research in that area. Based on the users’ selection of institution, another pane will display researchers at each institution working on the given topic; another pane shows their projects, another their communities of practice and so on. Users can also see where on the map these selected attributes are located geographically. Throughout use of the CAS, users are provided with a persistent overview of the domain from multiple views; information afforded is strongly associated with context.

4.2 Interaction Affordances

Beyond the overview afforded by the CAS UI, users can also determine the level of detail they wish to invoke for the attributes within the domain. We currently support two levels of zoom [28] into the detail to support user focus while maintaining current context. In addition to zooming, we also support user-determined domain reorganization [29]. We describe these below

First Level Zoom provides additional information for a selected attribute beyond the name of a particular instance. Selecting an element within an attribute panr provides an overview of that element. This information is populated in the Detail pane. For instance, clicking on a researcher in the Person pane will provide contact details about the researcher; selecting a project title will give project code, funding level. In this way users can quickly get at potentially all the information they may want without having to click out of their current context.

Second level Zoom If users wish more information on a particular element than the detail provides, by double clicking on an element in one of the lists, a new page will open with more information about that element. For instance, clicking on a researcher’s name opens up a dynamically generated web page for that researcher, including all papers, current stories about their research, community of practice details, and links to other projects. This *second level zoom* maintains a close relation to the current context.

Domain Reorganization Based on the users’ interests or requirements, they can reorient the domain to suit that locus[29]. If users wish to look at the CAS domain from the perspective of lprojects, funding, topic, regionl, or lregion, researcher, topl, these reorganizations are possible. In this way, the interaction allows users to reorient the domain on the fly to address questions as they evolve. In this way, the domain representation does not presume that users have queries formed clearly in advance,

but can evolve queries via the exploration of the domain.

Visualizers for Query Representation and Constraints In addition to the above techniques for text-based representations of data, we are developing multiple graphic visualizations to support domain overview, zoom and query constraints. The first of these is the Geography Visualizer. If users are interested in constraining their queries to only the north of England, for instance, dragging a reticule of 200km over that area of the map will result in any elements in the multicolumn lists being constrained by that range. Questions like are there correlations between region and funding levels become readily answerable, and can then be quickly followed with inquiries such as relation of funding to topic area or overlaps in community of practice. Similarly, results from queries have their geographical attributes populated in the map view. The Geography Visualizer is one of many possible graphic visualizers. Evaluations of the interface will help us determine which to bring online next.

With the *domain exploration* approach, and the associated interaction affordances described above, we have been able to provide for richer kinds of queries, which in turn, makes the enabling value of the triple store apparent.

4.3 Interaction from the System Level: Multiple Concurrent Query Example

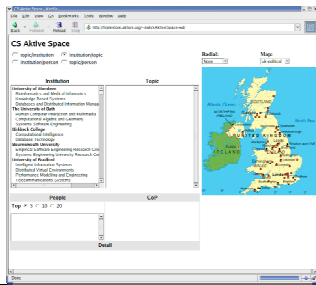
The current user interface for CAS is a dynamic XHTML document containing JavaScript code that formulates RDQL queries based on the users actions. JavaScript and XHTML was chosen as the UI platform so that it would work in any standards compliant web browser without additional software installation, and on most platforms. The RDQL queries are encoded as an HTTP request and sent to the 3store Apache module, which returns the results as an XML document, which is then parsed by the JavaScript to update the interface.

The JavaScript code also queries other semantic web services to perform other tasks, such as computing the CoP of an individual.

The user interface is designed such that actions on the user interface cause events to be triggered which fill out RDQL query templates based on the state of the user interface. In this way it is much like the Model/View/Controller paradigm [Krasner and Pope, 1988], with the knowledge base acting as the Model, the JavaScript as the controller and the HTML as the view.

A typical show interaction between the user interface and the triple store is shown below. The USING portion of the RDQL queries and some complete queries have been omitted for brevity.

Find all the research groups in the KB and the topics they study to build the initial UI

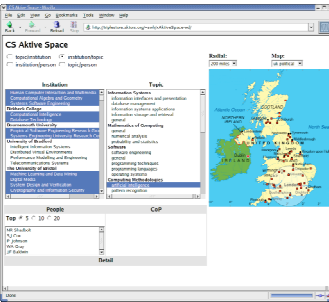


```
SELECT ?inst, ?topic, ?name, ?parent
WHERE (?topic, <rdf:type>, <acm:Research-Area>),
(?inst, <akt:has-research-interest>,
?topic),
(?topic, <rdfs:label>, ?name),
(?topic, <akt:sub-area-of>, ?parent)
```

```
[ 426 results]
SELECT ?uri, ?name, ?unit, ?uname
WHERE (?uri, <akt:has-research-interest>, ?ri),
(?ri, <rdf:type>, <acm: Research-Area>),
(?uri, <akt:has-pretty-name>, ?name),
(?uri, <akt:unit-of-organization>, ?unit),
(?unit, <akt:has-pretty-name>, ?uname)
```

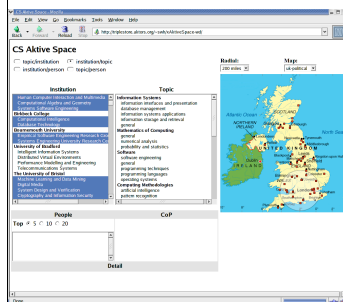
[516 results]

User selects south coast by placing reticule of map view over southern region of map



Topics are selected from the list from previous query

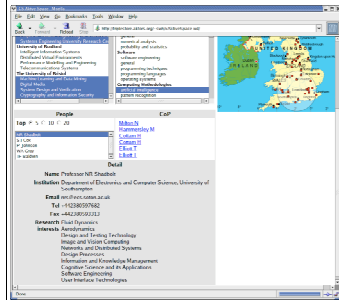
User selects the AI topic from the Topic pane



```
SELECT ?person, ?name, ?funds, ?unit, ?person2
WHERE (?proj, <akt:addresses-generic-area-of-interest>,
<epsrc:62>),
(?proj, <akt:has-project-leader>, ?person),
(?proj, <akt:has-funding>, ?funding),
(?funding, <akt:has-grant-value>, ?val),
(?val, <akt:has-amount>, ?funds),
(?person, <rdfs:label>, ?name),
(?person, <owl:sameIndividualAs>, ?person2),
(?person2, <akt:works-in-unit>, ?unit)
```

[repeated for every AI subtopic, 297 results total]

User selects person, make queries to fetch person metadata. Person information appears in the detail pane of the viewer



```
SELECT ?p, ?o
WHERE (<epscr:Person=14179>, ?p, ?ou),
      (?ou, <rdfs:label>, ?o)
```

[15 results]

```
SELECT ?p, ?o
WHERE (<epscr:Person=14179>, ?p, ?o)
```

[20 results]

5 Related Work

A number of applications on the Semantic Web (and its precursors) have informed the development of the CS AKTive Space and its ancillary technologies:

5.1 KA2 and Ontobroker

The KA2 Knowledge Acquisition Community Ontology [32] was an early example of an ontology-driven website designed to support a community. The content collected for the communal knowledge base was gathered almost entirely by the push model; members of the community were encouraged to annotate their web pages with information that would be collected and collated to form the resulting knowledge base. From the point of view of the knowledge acquisition process, the most important difference between KA2 and CAS lies in the granularity of data sources. In KA2, the use of annotated web pages as data sources yields a large number of small sources that must be maintained individually, although Staab et al do note that this is a time consuming and costly task. The KA2 portal adopts a task-neutral approach to their interface, as opposed to the task-specific domain exploration that we espouse in the CS AKTive Space; in the basic KA2 interface, queries are permitted on six axes (classes of object) with constraints from up to three axes applied conjunctively to the query.

While relatively successful as a proof of concept, the main failing of KA2 as a community endeavour was one of scale. The section of the knowledge acquisition community that took part was arguably too small for the effort to achieve critical mass and become self-sustaining, because the projected reward for participation (namely having consistent information to which one would not normally have access) was outweighed by the cost of participation in the short to medium term. In our development of CAS, we have tried to harness economies of scale and progress past this initial hurdle by choosing a broader subject base (so that there is a greater

likelihood that users will encounter information of which they were not previously aware, thus a greater reward), and by investing in the construction of an initial high-value knowledge base that provides an immediate reward to users and an incentive for further engagement.

5.2 SHOE

As with KA2, SHOE [21] relies on users annotating their web pages with relevant semantic markup that is harvested and aggregated in a communal knowledge base, although tools are provided for knowledge extraction from specific sources (e.g. CiteSeer). As a family of technologies rather than an application for a specific domain, SHOE supports the use of multiple ontologies, which is one of its particular strengths, but does not use the same knowledge representation formalism as current Semantic Web languages. The primary interface to the knowledge bases constructed from SHOE markup is the Semantic Search query tool, which although ontology-neutral, requires the user to specify the necessary characteristics (property values) of the objects to be returned by the search (the existence of more domain- and task-specific interfaces is mentioned, but not elaborated upon). In comparison, the CAS interface described in this paper aims to provide a user with a more flexible mode of interaction that eliminates the need to explicitly state the constraints on their query in this way.

5.3 Ontoport

The Ontoport project [13] is an ontological hypermedia system that provides a navigable interface to a knowledge base expressed in a given ontology. The data used in an Ontoport system is typically (but not necessarily) human-authored, and is presented as a browser interface. However, the content behind this hypermedia interface is static; the underlying infrastructure does not support query processing or inference, which results in a fixed page structure that provides only a single visualisation of the knowledge within. Our approach with CAS has been to provide task-specific visualisations of our gathered content, leveraged by the use of an RDF triple store with suitable query and inferential capabilities.

5.4 S-CREAM

S-CREAM [20] is a framework for creating metadata by applying annotations to web pages. S-CREAM avoids the maintenance bottleneck associated with semantic annotation by employing natural language information extraction techniques to semi-automate the annotation of documents. The tool used for this, Amilcare [16], produces a discourse representation for a document identifying the referents within the document that is then transformed into a set of annotations made against a given ontology. While the design of S-CREAM does automate much of the process of document annotation, the definition of the mapping from the structures in the discourse representation of the document to the ontological structures in the intended annotation is carried out manually, as are the equivalent mapping rules in our mediators.

5.5 GIS Systems

There are a number of Geographic Information System (GIS) based visual query systems that are superficially similar to CAS, with geographic range expression, there are however a number of fundamental differences, GIS database systems are often capable of resolving sophisticated geo-spatial queries based on topological expressions [12], while the CAS back-end has only basic number handling, as this is not a development focus. Conversely, GIS databases do not have the inferential or semantic representation capabilities of a knowledge base, as they are mainly concerned with conventional database retrieval. Equally their visual user interfaces are generally designed for query construction, rather than domain exploration [26]. Recent developments in this area include an integration of some knowledge base features [14], and ontologies have been used for some time for data integration [19].

6 Conclusions and Future Work: Social Issues and Technical challenges

It is our impression following the development of this interface that the inferential capability of RDF, combined with the structure provided by the ontology allows for comparatively simple queries to perform tasks that would require substantially more complex queries in an RDBMS system.

The CAS is offering a range of services, significant integration, and flexible interaction. We are dealing with millions of triples and yet it is still not enough. Despite our successful harvesting of the web and exploitation of various organizations' RDF, we still cannot reflect the richness of the queries our system can support because we do not always have enough content.

We have a strong proof of concept, but to get to the point where CAS becomes a meaningful application we need the content that we cannot get without participation. We need individuals and/or organizations to publish RDF content either directly against our ontology or else against an ontology we can translate. Participation cannot usually be enforced. Users have to see very strong benefits for the effort of publishing their content against our ontology.

We have started to see that happen within the UK CS community, and more recently, the eScience community. Interestingly, it is not simply the power of the back end tools that has provoked these enquiries and requests to apply CAS to other domains; it has been the user interaction. There is an immediate appeal to the fact that patterns and gestalts, particular and general content exploration can be so rapidly effected.

Many of us will have been frustrated by the fact that so often individuals and organizations, our managers and colleagues ask us for the same basic content in different formats and configurations. One of the attractions of mediating content through shared ontologies is that re-presentation and repurposing of content becomes so much easier. A huge collective effort is made in the UK every few years to collect content about the state of research in Higher Education. This so-called Research Assessment Exercise is important since the content collected goes on to form the basis of rankings of Department to particular grades. These grades are direct drivers on the

funding departments receive. The clerical effort of collecting and make available this content is many hundreds of person years. Much of this effort is the equivalent of the semantic clerk – someone who has to work out how to integrate the various content into the formats required for the particular exercise. These sorts of requirement can provide an opportunity to convince communities and the organizations demanding the content that that for a little investment in Semantic Web engineering and ontology construction they could dramatically reduce the cost of repeated content collection.

One other issue, behind the content and behind the UI is service integration. One of the ways we have been dealing with the issue in AKT is, for the time being, ignore service discovery and focus on service integration with AKT partner technologies. Partners are working to integrate their services with the central triple store. What this means is that CAS users have a variety of services they can invoke that we have evaluated as relevant to working with CAS. Through the process of working through these integration problems we plan to have something to report about how these specific integration efforts can be generalized towards a loose protocol so that integration of dynamically discovered/requested services can be supported.

In the future we hope to include more component services. All the services have to be able to do is interrogate the triple store and be able to take advantage of the ontological structures that provide our common models for content integration. We already have web services that can classify our research papers against our research area ontology. We are able to invoke a natural language learning service that is able to detect ontologically significant parts of texts and abstracts – titles, topics, authors and general term entity recognition. We are incorporating a dynamic link service and intend to provide a range of e-scholar services such as detecting parts of the research area ontology that are rich in publications with various types of impact factor. One ambition is that in future RAE's the panels will not just have standard bibliometric evidence, number of citations and impact factors of journals. With a system like CAS we will be able to engage in semiometrics – looking at a whole range of measures and content implicit in the Conventional Web and meta content explicit in the Semantic Web to build much richer views as to the real research activity underway in our discipline.

Acknowledgments

This work was supported under EPSRC grant GR/N15764/01.

References

1. 3store: An Rdf Triple Store. <http://sourceforge.net/projects/threestore/> 2002.
2. The Akt Reference Ontology. <http://www.aktors.org/publications/ontology/> 2003.
3. Granite Nights: An Agent-Based Evening Scheduler for the Granite City. <http://www.csd.abdn.ac.uk/research/AgentCities/compo/>. 2002.
4. Hyphen: An Information Source for Uk Researchers,. <http://hyphen.info/>

- 2001.
5. The Meet-O-Matic Meeting Scheduler, <http://www.meetomatic.com> 2002.
 6. *Scenario Based Design*. John Wiley, 1995.
 7. Abernethy, N.F. and Altman, R.B., Sophia: Providing Basic Knowledge Services with a Common Dbms. In *Proceedings of the 5th International Workshop on Knowledge Representation Meets Databases (KRDB '98): Innovative Application Programming and Query Interfaces*, (Seattle, Washington, USA,, 1998).
 8. Ahlberg, C. and Shneiderman, B., Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays. In *Conference on Human factors in computing systems: celebrating interdependence (CHI94)*, (Boston, Massachusetts, USA, 1994).
 9. Alani, H., Dasmahapatra, S., Gibbins, N., Glaser, H., Harris, S., Kalfoglou, Y., O'Hara, K. and Shadbolt, N., Managing Reference: Ensuring Referential Integrity of Ontologies for the Semantic Web. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW'02)*, (2002).
 10. Alani, H., Dasmahapatra, S., O'Hara, K. and Shadbolt, N. Identifying Communities of Practice through Ontology Network Analysis. *IEEE Intelligent Systems*, 18 (2). 18-25.
 11. Berners-Lee, T., Fielding, R. and Masinter, L. Uniform Resource Identifiers (Uri): Generic Syntax. *Internet Engineering Task Force*, RFC2396, 1998,
 12. Cai, G., Geovibe: A Visual Interface to Geographic Digital Library. In *ACM-IEEE JCDL Workshop on Visual Interfaces*, (2001).
 13. Carr, L., Kampa, S. and Miles-Board, T. Metaportal Final Report: Building Ontological Hypermedia with the Ontoportel Framework . *Electronics and Computer Science, University of Southampton*, 6976, 2002,
 14. Casey, M.J. and Austin, M.A., Semantic Web Methodologies for Spatial Decision Support. In *International Conference on Decision Support Systems in the Internet Age*, (Cork, Ireland, 2002).
 15. Ciravegna, F. Designing Adaptive Information Extraction for the Semantic Web in Amilcare. in Handschuh, S. and Staab, S. eds. *Annotation for the Semantic Web*, IOS Press, Amsterdam, 2003 forthcoming.
 16. Ciravegna, F., Dingli, A., Guthrie, D. and Wilks, Y., Mining Web Sites Using Unsupervised Adaptive Information Extraction. In *10th Conference of the European Chapter of the Association for Computational Linguistics*, (Budapest, Hungary, 2003).
 17. Domingue, J. and Motta, E. Planetonto: From News Publishing to Integrated Knowledge Management Support. *IEEE Expter*, 15 (3).
 18. Eisenstadt, M. and Dzbor, M., Buddyspace: Enhanced Presence Management for Collaborative Learning, Working, Gaming and Beyond. In *JabberConf Europe*, (Munich Germany, 2002).
 19. Hakimpour, F. and Timpf, S., Using Ontologies for Resolution of Semantic Heterogeneity in Gis. In *4th AGILE Conference on Geographic Information Science*, (Brno, 2001).
 20. Handschuh, S., Staab, S. and Ciravegna, F., S-Cream: Semi-Automatic Creation of Metadata. In *13th International Conference on Knowledge*

- Engineering and Knowledge Management (EKAW'02)*, (2002).
21. Heflin, J. and Hendler, J. A Portrait of the Semantic Web in Action. *IEEE Intelligent Systems*, 16 (2). 54-59.
 22. Leonard, T. and Glaser, H., Large Scale Acquisition and Maintenance from the Web without Source Access. In *First International Conference on Knowledge Capture (K-CAP2001)*, (2001).
 23. Manola, F. and Miller, E. Rdf Primer. <http://www.w3.org/TR/rdf-primer> 2003.
 24. Niles, I. and Pease, A., Towards a Standard Upper Ontology. In *2nd International Conference on Formal Ontology in Information Systems (FOIS-2001)*, (2001).
 25. Norrie, M.C., Reimer, U., Lippuner, P., Rhys, M. and Schek, H.J., Frames, Objects and Relations: Three Semantic Levels for Knowledge Base Systems" Reasoning About Structured Objects: Knowledge Representation Meets Databases. In *1st Workshop KRDB'94*, (Saarbrücken, Germany, 1994), 20-22.
 26. Olston, C., Stonebraker, M., Aiken, A. and Hellerstein, J.M., Viquing: Visual Interactive Querying. In *IEEE Symposium on Visual Languages*, (1998).
 27. Partridge, C., The Role of Ontology in Semantic Integration. In *Object-Oriented Programming, Systems, Languages, and Applications - 11th Workshop on Behavioral Semantics*, (2002.).
 28. Schaffer, D., Zuo, Z., Greenberg, S., Bartram, L., Dill, J., Dubs, S., Roseman and M., N. Navigating Hierarchically Clustered Networks through Fisheye and Full-Zoom Methods. *ACM TOCHI*. 162 – 188.
 29. schraefel, m.c., Karam, M. and Zhao, S., Mspace: Interaction Design for User-Determined Domain Exploration. In *International Workshop on Adaptive Hypermedia*, (Nottingham, UK, 2003).
 30. schraefel, m.c., Karam, M. and Zhao, S., Preview Cues for Exploring Domain Hierarchies. In *Interact 2003*, (Switzerland, 2003, forthcoming).
 31. schraefel, m.c., Zhu, Y., Modjeska, D., Wigdor, D. and Zhao, S., Hunter Gatherer: Interaction Support for the Creation and Management of within-Web-Page Collections. In *Proceedings of the eleventh international conference on World Wide Web*, (Hawaii, 2002), 172-181.
 32. Staab, S., Angele, J., Decker, S., Hotho, A., Maedche, A., Schnurr, H.-P., Studer, R. and Sure, Y., Semantic Community Web Portals. In *9th International World Wide Web Conference*, (2000).
 33. Vardi, M.Y., On the Complexity of Bounded-Variable Queries. In *Fourteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, (San Jose, California, 1995).