

Collaborative Semantic Web Browsing with Magpie

John Domingue, Martin Dzbor and Enrico Motta

Knowledge Media Institute, The Open University, Milton Keynes, UK
{J.B.Domingue, M.Dzbor, E.Motta}@open.ac.uk

Abstract. Web browsing is often a collaborative activity. Users involved in a joint information gathering exercise will wish to share knowledge about the web pages visited and the contents found. Magpie is a suite of tools supporting the interpretation of web pages and semantically enriched web browsing. By automatically associating an ontology-based semantic layer to web resources, Magpie allows relevant services to be invoked as well as remotely triggered within a standard web browser. In this paper we describe how Magpie trigger services can provide semantic support to collaborative browsing activities.

1 Introduction

Magpie is a tool supporting the *interpretation of web pages*, acting as a complementary knowledge source, which a user can call upon to gain instantaneous access to the background knowledge relevant to a web resource. Magpie automatically associates a *semantic layer* to a web resource, rather than relying on a manual annotation [8]. Semantic layering relies on the availability of an *ontology* [12] – an explicit, declarative representation of a domain. Ontologies help to assign meaning with the information on a web page, and on the basis of the identified meaning, to offer the user several functionalities based upon semantic services.

The semantic layers produced using the Magpie framework transform an arbitrary web document into a semantically (or contextually) *enriched document*. In previous projects we have developed methodologies for enriching documents with both informal and formal annotations to support organizational learning [19, 21]. Our Enrich methodology was successfully applied in domains as diverse as aerospace design engineering and widening participation in education. Magpie extends the Enrich framework through the on-the-fly semantic layering of web documents [6].

In earlier papers [6-8] we gave an overview of the Magpie framework and design principles and outlined Magpie's basic functionalities. A key principle underlying the design of Magpie is that the ontology provides an explicit interpretative viewpoint over the web resources. Since different readers of a particular web resource (e.g. a personal home page) will most likely have different degrees of familiarity with the information shown in a web page and with the relevant background domain, they will necessarily require different levels of sense-making support. Hence, the semantic layering and consequently support for semantic browsing using the Magpie framework needs to be sufficiently *flexible* and *extendible* to allow for the different users' needs.

In this paper we discuss how Magpie can support collaborative semantic web browsing. More specifically, how users can be informed when a colleague finds information in a web page which is relevant to the current user's context. Hence, the framework that has been designed to support an individual user in interpreting web resources can also facilitate community-based knowledge and experience sharing. The economic benefits of 'group-wide interpretation' are gained by providing faster access to the relevant expertise and experience related to the semantically relevant topics or issues *within the group* or community.

The rest of this paper is structured as follows. Section 2 describes a small scenario where Magpie supports collaborative web browsing. We then describe the overall architecture of Magpie in some detail. A review of related work in section 4 is followed by some conclusions.

2 A Scenario of Magpie in Use

Imagine commercial company that specializes in finding interesting technologies and exploiting them for profit. Two members of the company, Gill and Martin, are browsing the web site of the Knowledge Media Institute (KMi) with a goal of finding potentially interesting projects.

Gill brings up her web browser and selects a particular ontology, in this case the AKT Reference Ontology¹, to act as a point of view for browsing. Gill then subscribes to a number of collectors – a specific type of Magpie trigger service. The interface for trigger service subscription is shown in Fig. 1. Using the interface Gill subscribes to the People, Project and People's Projects collectors. Gill also uses the Magpie preferences dialog to indicate that Martin is a co-worker involved in a shared information gathering task.



Fig. 1. A screen snapshot of the Magpie interface for subscribing to services

Gill first looks at the home page for the Alice project, shown in Fig. 2, and uses the Magpie toolbar to highlight key ontological concepts on the page; shown in close up in Fig. 3. As can be seen, Magpie preserves the appearance of the page, and offers a toolbar-based user interface to highlight concepts of a particular type. The buttons in the Magpie toolbar are ontology dependent; i.e. they correspond to the selected classes from the specific ontology the user selected. For the AKT reference ontology

¹ <http://www.aktors.org/publications/ontology/>

the key concepts are research area, project, person and organisation. Selecting a button in the toolbar highlights the associated concepts within the page. Thus, the toolbar embodies our rationale to base the interpretation of a web resource on a particular viewpoint. Because the viewpoint to be taken will depend on the specific user's task we leave the choice of ontology up to him or her.

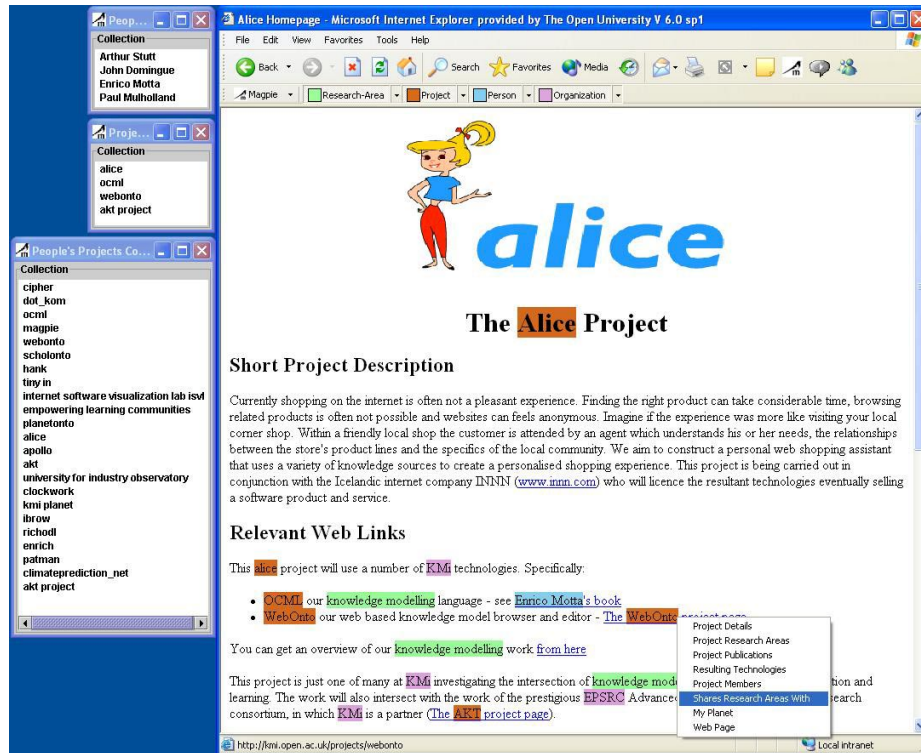


Fig. 2. The Alice project home page viewed through Magpie. Known *people*, *organizations*, *projects* and *research areas* are highlighted using the Magpie toolbar (shown in close-up in Fig. 3). On the left-hand side are three Magpie collectors – the top two log the people and projects found in the browsing session. The bottom one shows the (not explicitly mentioned) projects associated with the people found.

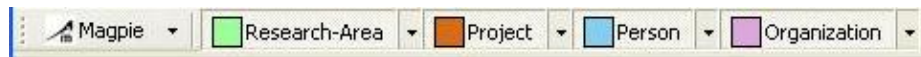


Fig. 3. A close-up view of the Magpie toolbar. Each item in the toolbar corresponds to a particular concept in the selected ontology. Selecting a button in the toolbar highlights the associated terms within the current web page.

In Magpie, each annotated and highlighted concept becomes a ‘hotspot’ that allows the user to request a menu containing relevant functionalities for that item. In Fig. 2, the contextual right-click on the ‘WebOnto’ concept reveals a menu of semantic services for projects. The choices generally depend on the ontological classification of a

particular concept within the selected ontology. Some Magpie functionalities are straightforward (e.g. ‘Project Details’). Others are more sophisticated enabling the user to explore the semantic neighbourhood of a particular concept, and to navigate to the web resources that are semantically related but not explicitly linked to. For example, the service ‘Shares Research Interests With’ responds with a list of projects having common research interests with the WebOnto project. The list as shown in Fig. 4 cannot be found anywhere on the KMi web site. However, the interpretative viewpoint of the selected ontology gives the reader an insider’s view. Additionally, the answer to a semantic query, which is a web resource, can be further browsed semantically. Thus, Magpie merges the independent mechanisms for recognizing semantic relevance and browsing web resources.

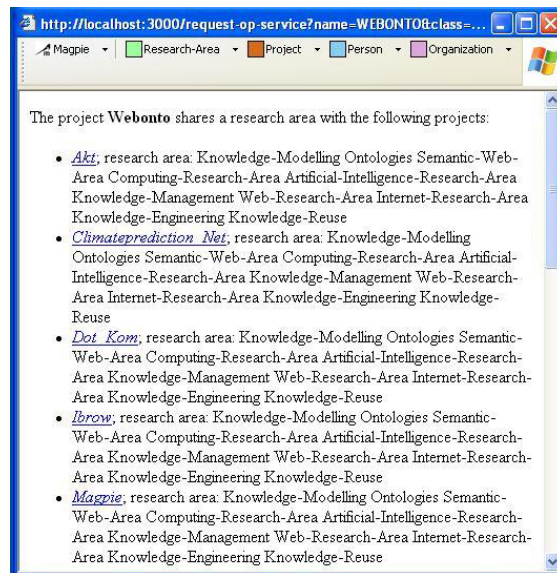


Fig. 4. Results of the semantic query ‘Shares Research Areas With’ invoked for the ‘WebOnto’ project by the semantic menu action depicted in Fig. 2. Each bullet shows the name of a project followed by a list of overlapping research areas.

The left-hand side of Fig. 2 shows three Magpie collectors. These are automatically filled by Magpie trigger services, which use a semantic log knowledge base (KB) that keeps a record of a browsing session and the entities found on the web pages. Collectors show a semantically filtered view of the semantic log providing a structured record of user’s browsing history. Moreover, all visited pages are semantically annotated with a specific ontology rather than merely indexed by keywords. The top two collectors in Fig. 2 record the people and projects recognized on this and previously visited web pages thus serving as a type of semantic bookmark.

The bottom collector lists those projects that are relevant to the discovered people but were not mentioned explicitly on any visited page. This collector represents what we term a semantic neighbourhood. Items within a Magpie collectors offer the same functionalities as hotspots on a web page; it is possible to right-click them and request appropriate semantic services. Other types of trigger services are discussed in [7].

Now Gill moves to the KMi Stadium project page. As she does a Magpie Contact window pops up. This notifies Gill that her colleague, Martin, was recently, looking at a related technology, Lyceum, which has the same author and owner. The (green) icon to the right of Martin's picture indicates that he is currently online within an Jabber-based instant messaging client, BuddySpace [9], developed in our lab. Selecting the 'Contact' button in the window brings up an instant messaging dialog window.

Note that the above interaction relies on the existence an ontology which specifies the relationships between project, technologies, organisations and people and which has been populated by the appropriate entities.

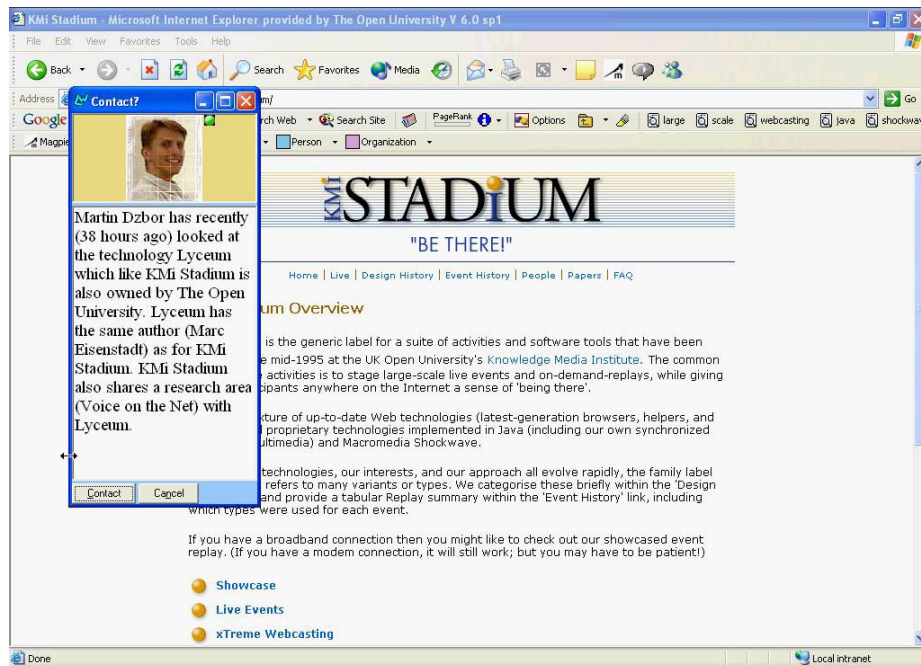


Fig. 5. A screen snapshot showing the collaborative browsing support provided by Magpie. The 'Contact?' window shows that a colleague, Martin, has been browsing a similar page.

3 The Magpie Architecture

In the introduction we stated that a key design principle underlying Magpie was to allow the users who normally have little knowledge of a particular discourse, to browse web resources and take the full advantage of relevant contextual knowledge. This broad aim was elaborated into a set of design requirements we believe are important for designing user-oriented and user-driven applications for browsing and understanding the semantic web. The full list of requirements is given in [8]; here we highlight the key points:

- Magpie should offer semantic web functionality with near-zero additional costs for the users;
- Magpie should provide access to semantic information using a ‘one-click’ (or less) principle.

We will now describe how the Magpie architecture supports the interactions described in the scenario. The architecture uses an infrastructure that we have developed, called IRS-II [20], which supports the publishing and invocation of semantic services. A key feature of IRS-II is that it is based on the UPML framework [11] and therefore differentiates between tasks, problem solving methods (generic reasoners) and domain models. The split between tasks and problem solving methods enables us to clearly separate the activities of implementing semantic services from making them available in a fashion where they can be easily invoked with little or no overhead to the end-user. We will not say any more about the IRS-II in this paper but will instead focus on the aspects of the architecture related to collaborative web browsing.

3.1 An Overview of the Magpie Architecture

The Magpie architecture (shown in Fig. 6) is essentially that of a mediator. The mediation occurs between the user browsing web pages and the suite of remotely accessible semantic services. Some services are from the ‘user-requested’ category, while others are ‘triggered’. This difference will become obvious in the following sections.

The Magpie architecture consists of a *Service Provider* and a *Service Recipient*. This emphasizes the fact that the document source (i.e. web server) and the source of a semantic service may not be on the same machine. Similarly, a concept of ‘client’ suggests that an active role is always on the side of a browser with a server passively serving requested data. In the case of our trigger services, it is the server component that drives the conversation to provide information that is semantically relevant but not explicitly requested by the user.

Currently, the Service Provider component of Magpie is built around a suite of tools providing access to a library of domain ontologies, populated KBs, ‘hand-crafted’ semantic services, and a semantic log KB. This library accepts ontologies represented in RDF(S) [1], DAML+OIL [5], and OCML [18] (the latter being the internal representation we use for reasoning). An OWL [23] import/export mechanism is nearing completion. Most of the ontologies relate to a particular domain of discourse (e.g. academic organizations or design).

3.2 Service Recipient Components

The Magpie components on the service recipient (‘client’) side comprise the Magpie Browser Extension (of Internet Explorer), the Magpie Service Dispatcher, and Trigger Service Interfaces. The Magpie Browser Extension is embedded in the browser as a plug-in and controls the interaction with Magpie facilities. The plug-in preserves as much as possible the appearance of a web page before semantic annotation through the application of semantic layers. Its user interface components visualize entities

found in a web page, and enables users to interact with the semantic services through contextual menus.

The Browser Extension also contains a simple parser that annotates the entities from a particular ontology on a web page using an ontology-derived lexicon. Although not as complete as some grammatical named-entity recognition (NER) techniques, the lexicon-based parsing is extremely fast, and thus satisfies our near-zero overhead constraint. While lexicon-based NER is fast, its major drawback is precision and completeness. We are currently working on merging a lexicon-based and a grammar-based NER into the core Magpie plug-in. However, in order to avoid adding ever more complex NER techniques to the browser extension, we also experiment with implementing advanced NER algorithms as (semantic web) services a user would be able to subscribe to.

When evaluated using standard language engineering measures such as precision of recognition and recall, Magpie exhibited a high degree of precision within the domain. When tested outside of the intended domain, the precision fell dramatically, due to the brittleness of the fast lexicon-based NER techniques that the plug-in uses. In our tests, we used Magpie and benchmarked it against GATE [4] and a pattern-based NER tool called eSpotter developed in-house. The NER technique used in Magpie outperformed those in the benchmarked competitors in terms of timing and almost 100% precision of recognition whilst within the tested domain. This satisfies our main criterion to provide NER-based functionality with zero time overhead. The obvious drawback is the quick degradation of recognition performance when leaving the domain for which the lexicon was created.

The *Magpie Service Dispatcher* acts as a dedicated semantic proxy for the user's web browser. It manages the communication between the Browser Extension and the appropriate dispatcher of the service providers. The Magpie Dispatcher delivers user's requests and the provider's responses using customized XML messages. *Collectors* are one form of Magpie Trigger Service Interface that display data pushed by the applicable trigger services.

The Magpie semantic layer comprises customized <SPAN...> tags enclosing named entities linking them with the relevant instances and classes within the chosen ontology. This approach to visualizing semantic layers gives users the control of what types of entities are visible at any time. This improves navigation through the (semantically enriched) content, and avoids overwhelming the users with too much information, which is a problem with some link-recommending systems (see the discussion in [17]). When the entities are recognized, the Dispatcher passes them to a *Central Magpie Service*, which asserts the entities as facts into the *Semantic Log KB*. The purpose of and mechanisms within semantic logging are addressed later.

In addition to highlighting entities and offering semantic service menus, the Magpie infrastructure supports *Trigger Service Interfaces*. These comprise various summarizers, recommenders or collectors. Unlike contextual, menu-based services, the user does not explicitly request a trigger service but subscribes to it (see Fig. 1). When a trigger service is activated information is pushed from the service provider to reflect particular patterns in the log of recognized entities. Nonetheless, a user might not be interested in receiving messages from all trigger service providers.

Collaborative browsing is supported via the trigger service interfaces. The messages are instigated by the browsing activity of registered collaborators.

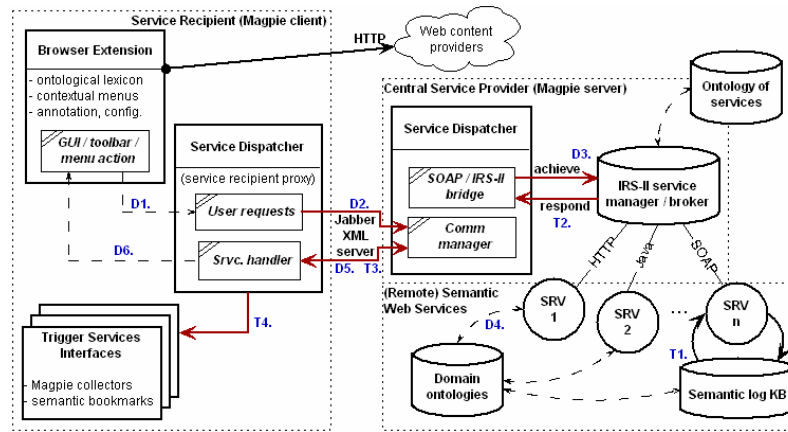


Fig. 6. Schematic architecture of the Magpie framework.

3.3 The Magpie Service Dispatcher

The role of the Service Dispatcher is to handle all interactions between the user, the Magpie-enabled browser and the respective Magpie service providers. In principle, the dispatcher is an alternative to the GET/POST requests available in standard HTTP protocols. Although Magpie supports HTTP requests, a growing number of services are available in formats that do not lend themselves to a seamless integration into a standard web browser. Separating interface and communication gives us several advantages.

The dispatcher abstracts the user's action (e.g. a click selection of a semantic service), and automatically translates this into the appropriate XML-based request to the service provider. Generally, each request may have a specific XML form or template associated with it, and this is automatically 'filled in' by Magpie before being dispatched to the service provider. The dispatcher delivers the XML-coded request on the user's behalf.

This leads to a second benefit. Since the Magpie dispatcher acts on behalf of the user and can be identified as such, it is possible to implement the service request/response communication in an *asynchronous* way. In other words, the communication stream does not have to remain open while the service providers are waiting for the response from the actual service execution. When the response is ready, the service provider can easily send it back to the user who originally requested it (or better to a dispatching agent acting on behalf of the user).

Finally, the asynchronous capability attached to the standard web browser is critical for supporting what we labelled as trigger services. People may be familiar with numerous pop-up windows that are 'pushed' to the user's desktop while browsing. However, the principle of trigger services is totally different. They are not designed for 'blanket coverage' of all users browsing a particular page. They enhance a particular viewpoint facilitated through a particular ontology, and they are only activated, when a specific pattern emerges among the entities on a page. Therefore, the dispatcher prevents 'spam', and enables real push services.

Some of the messages coming from trigger services may be handled directly by the dispatcher thus lifting the burden from the actual web browser. The trigger messages may be re-directed to a more appropriate user interface than a plain web browser window (e.g. a graphic visualization or tree-based collector widget). However, the important feature is that the presence of a dispatcher enables a service provider to address

users. This type of *two-way communication* is critical to supporting collaborative browsing and is not possible with standard HTTP based protocols.

Bi-directional information exchange becomes useful when we allow for the negotiation of the ontology to be used for sense-making. In principle, a user may refer to different ontologies to interpret a single web page. Some ontologies may be remote, and they may be defined in different formats. One of the dispatcher based functionalities that we are currently experimenting with is to create an import facility for various key web KR formats (RDF, OWL, etc.). Such an ‘ontology-to-lexicon translator’ might also allow the user to customize an ontology or select only classes that are needed.

3.4 The Semantic Log and Collaborative Trigger Services

The key to collaborative browsing is the availability of a *shared semantic log*. The URL of each web page browsed is passed, via the service dispatcher, to a web service which downloads the document and extracts the items which match the ontology derived lexicon. The items are then asserted into a semantic log KB as facts. Magpie *watchers* monitor and respond to patterns in the asserted facts. When the relevant assertions have been made for a particular watcher, a semantic service response is *triggered*, and applicable information delivered to the dedicated window in the user’s web browser. An example Magpie watcher definition is given below.

```
(def-watcher collect-peoples-projects peoples-projects
  (found-item ?time ?address ?page-url ?person ?user-id)
  (person ?person)
  (has-project-member ?project ?person)
  (not (found-item ?time2 ?address ?page-url2 ?project))
  :action (collect :peoples-projects ?user-id ?project
                  ?page-url ?address)))
```

Fig. 7. Watcher definition for trigger service “People’s Projects” (in OCML). The results of using this method on the semantic log KB are shown in the bottom left-hand collector in Fig. 2.

```
(def-watcher alert-collaborator-for-similar-technology contact
  (found-item ?time1 ?address1 ?page-url1 ?technology1 ?user-id1)
  (found-item ?time2 ?address2 ?page-url2 ?technology2 ?user-id2)
  (within-3-days ?time1 ?time2)
  (are-related-technologies ?technology1 ?technology2)
  (not (= ?user-id1 ?user-id2))
  (not (= ?page-url1 ?page-url2))
  :action
  (alert-collaborator :similar-technology ?user-id1
                     ?user-id2 ?technology1 ?technology2)))
```

Fig. 8. Watcher definition for a collaborative trigger service “People’s Projects” written in OCML. The results of using this method with the semantic log KB are displayed in the bottom left-hand collector in Fig. 2.

Collaborative trigger services make use of the fact that users working together will share a semantic log. We can then create watchers which trigger services for collaborators. The definition of the watcher for the alert shown in Fig. 7 is shown in Fig. 8. This definition notifies a user when a collaborator has recently looked at a web page

containing a related technology. In this context technologies are related if they cover related research areas, have the same author or are owned by the same organization.

We know from the studies of Tauscher and Greenberg that on average 58% of visits to web sites are revisits. We do not know if any studies which extend this work for collaborating teams but we presume that colleagues engaged on similar information gathering tasks would come across the same pages. Watchers such as those shown in Fig. 8 prevent the duplication of work and facilitate team knowledge sharing.

4 Related Work

To our knowledge, one technology that strongly resembles Magpie (at least in respect to the appearance) is the KIM plug-in for Internet Explorer. KIM [22] is a Knowledge and Information Management platform for automatic semantic annotation, web page indexing and retrieval. Similarly to Magpie, KIM uses named entities as a foundation of a document's semantics, and assigns ontological descriptions and definitions to the entities in the text. The platform takes advantage of a massive populated ontology containing common 'upper-level' concepts (e.g. locations, dates, organizations or money) and their instances.

Unlike Magpie, the essence of KIM is a NER technique based on the GATE platform [4] but extending the typically flat set of NER rules with ontological hierarchies. The entities are recognized in the text by the KIM server API, and in parallel they are associated with respective instance in the ontology. Here, GATE provides a lot of support in recognizing abbreviations, incomplete names and co-referenced terms. All these functionalities enable KIM to recognize both already-known and new named entities. While the actual annotation is similar to that in Magpie, the generic capabilities of GATE offer one important benefit – KIM is able to extend its knowledge base by inserting newly recognized entities as instances of appropriate ontological classes.

The major difference between Magpie and KIM is the 'aftermath' of recognition. KIM enables the user to explore the semantic definitions/descriptions of the recognized items in a semi-structured format. Rather than showing complex semantic definitions, Magpie allows the user to choose which service s/he is interested in. This enables Magpie to use the reasoning capabilities of ontologies to a much greater extent, and eventually provide a more customized and open-ended user interaction experience. To summarize, we see Magpie and KIM as two complementary tools catering for different user needs.

One of the inspirations for Magpie was the COHSE family of systems [3]. COHSE combines an Open Hypermedia System with an ontology server into a framework for ontological linking. As with Magpie, it uses an ontology-derived lexicon; but instead of annotations it adds links to related web pages. The links are added either by a proxy server or via an augmented Mozilla™ browser. The distinctions between Magpie and COHSE are due to their differing design goals. The goals for COHSE were (i) to separate web links from the web pages and (ii) to make the links conceptual (i.e. potentially generated from an ontology). The goal for Magpie is to support interpretation and information gathering by tapping into relevant conceptual knowledge. Magpie's interface enables ontological differences to be highlighted, and the services

provided are dependent on the class of entity found. Magpie also offers trigger services via semantic logs. Neither type of Magpie service is designed to replace traditional links rather they act as an auxiliary knowledge source.

A host of annotating tools and systems have emerged in recent years. For instance, the Amaya editor implementing the Annotea framework [15], the CREAM-based Ont-O-Mat/Annotizer [13], MnM [24], and the SHOE framework [14]. The main difference between these efforts and the approach taken by Magpie and KIM lies in the origin of the annotations. The above tools relied (at least partly) on the manual annotation of a document. Once the annotation was seeded by the user/librarian, the annotation systems typically provide tools to take over from the user and automate the process. For instance, Ont-O-Mat uses the rule learning capability of Amilcare – a tool often used in conjunction with GATE.

Magpie differs from these tools by intertwining entity recognition, annotation, and a formal representation of an ontology. Our approach to entity recognition and annotation results in a near-zero overhead for the users (including document authors) who are not required to annotate their work. Ontologies, on the other hand, provide a shared (and formal) understanding that is more likely to outperform ‘free-text’ annotations in conveying the same meaning to different users. We admit though that free-hand annotations are superior in terms of personalized and customized interpretation.

Another strand of research that is relevant to the Magpie framework is Letizia [16] and its idea of a reconnaissance agent. This type of agent “looks ahead of the user on the links on the current web page”. The authors argue that pre-filtering the web pages may improve the relevance and usefulness of browsing. The functionality similar to that of Letizia (“logged entities reconnaissance”) is implemented in Magpie through its semantic logging and ontological reasoning upon the semantic log KB. In [7] we describe a Magpie component which visualizes the semantic neighbourhood of a browsing session through the semantic log.

There are a number of systems which support collaboration through augmented web browsing. The CWB (Collaborative Web Browsing) [10] is one of several systems which allows users to synchronously browse the same web page. One of the novel features of CWB is that it is extremely lightweight and does not rely on a plug-in, binary executable or Java applet. The I2I system [2] allows users to leave calling cards on web pages which state that the user would be happy to talk to other users on a topic related to the web page contents. To our knowledge there are no collaborative web browsing tools which utilize a shared ontology.

5 Discussion and Conclusions

In this paper we described the generic Magpie framework for semantic web browsing focussing on how trigger services can support collaborative sense making on the web. Our infrastructure supports colleagues working together on related information gathering activities. Users are notified about relevant browsing activities of co-users.

Supporting semantic browsing requires a number of key technologies. An ontology which captures a shared viewpoint enables inferences to be made as to when the contents of one web page are related to another. It is vital that the ontology is automati-

cally populated from relevant web resources. Our currently deployed version, which relies on the AKT Reference ontology, uses a mixture of ODBC database interfaces and generic scripts for population – see [6] for more details.

In this paper we also presented a brief summary of our evaluation study, where we benchmarked NER techniques used in Magpie with those used by GATE and another in-house developed NER tool. We concluded that Magpie exhibited superior performance in terms of time needed to process web pages (accelerating NER by as much as factor of 10-15 compared to GATE). Moreover, with such a low time overhead Magpie achieves near-100% precision *within* the domain for which lexicon was intended. The major ‘drawback’ of the lexicon based technique used in Magpie is a rapid degradation in precision and recall when moving *outside* of the intended domain. However, a complementary tool (eSpotter) which uses patterns and grammars has been tested in conjunction with Magpie, and the envisaged merger of the two technologies promises to overcome the brittleness of ontologically derived lexicons for NER.

A shared (ontology based) semantic log captures knowledge about the interesting entities found by a community of users browsing the web. Pro-active trigger services can be fired whenever interesting patterns of communal behaviour are detected. These patterns can of course indicate overlaps indicating users are looking at related web resources or disjointness indicating that users are looking at complementary web resources. The triggering pattern will naturally depend on users, their task and the type of collaboration. In many settings users will only wish to share a strictly limited portion of their browsing activity.

The final key technology is provided by the Magpie dispatcher which enables a user’s browser to be addressed directly. Standard HTTP does not support client addressing, nor the pushing of information to the user in a structured and controlled manner. The Magpie service dispatcher provides this complementary functionality, which in turn is essential for a meaningful distribution of trigger services. Moreover, the messages are passed between the user’s dispatcher and the provider’s dispatcher before they reach the actual service. The dispatcher also supports the login/password-based authentication of the users, which enables the user to select or subscribe to only to those services s/he is authorized to.

The Semantic Web promises to deliver a system where computational resources can be brought to bear to aid human users in knowledge intensive tasks. Through a combination of the technologies outlined above Magpie is able to link collaborators engaged in shared browsing activities through formally specified relations. Tim Berners-Lee argues that “Semantic Web is about integration of resources rather than browsing the resources”². What is the role of Magpie or Magpie-like technologies that inherently address the issues of *browsing*? While Magpie might not be “*the* semantic browser”, it definitely is “a browser for the Semantic Web”. In other words, Magpie acts as an *integrator* residing within the user’s web browser that facilitates access to the relevant *distributed* knowledge resources.

The Magpie framework is not intended for the purposes of browsing the Semantic Web. Its main contribution to the Semantic Web research community is in allowing users to browse the standard Web whilst utilizing the concepts and relationships cap-

² Quoting Tim Berners-Lee’s keynote speech delivered at 2nd International Semantic Web Conference, Sanibel Island, Florida, US, October 2003.

tured within a selected ontology. We emphasize that majority of semantic services (whether on-demand or triggered) that are offered through the Magpie framework are published using the standards of semantic web services. Consequently, Magpie behaves more like an end-user interface for accessing and interacting with these distributed services rather than a “browser”.

Furthermore, a significant amount of knowledge is available on the standard Web, which is not annotated, and typically coded in a semantically ‘flat’ format. Another contribution Magpie offers is the utilization of this content and its enrichment using a semantic layering based approach. We believe that this is an important step towards supporting semantically enriched browsing both by individuals and communities.

One of our case studies conducted in the context of an Open University climatology course gives students (beginners!) access to state-of-the-art research findings resulting from the largest climate modelling exercise – *climateprediction.net*. Without Magpie’s ability to link these highly complex findings with the course concepts, the students would need to spend considerable effort in terms of study time on consulting their tutors and various traditional sources (e.g. course books). Therefore, we believe that existence of a tool supporting semantically enriched browsing is an important application for early adopters and users of Semantic Web technologies.

Acknowledgements

The Magpie research is supported by the Advanced Knowledge Technologies (AKT) and *climateprediction.net* projects. AKT is an Interdisciplinary Research Collaboration (IRC) sponsored by the UK Engineering and Physical Sciences Research Council by grant no. GR/N15764/01. The AKT IRC comprises the Universities of Aberdeen, Edinburgh, Sheffield, Southampton and The Open University. *Climateprediction.net* is sponsored by the UK Natural Environment Research Council and UK Department of Trade e-Science Initiative, and involves Oxford University, CLRC Rutherford Appleton Labs and The Open University.

References

- [1] Brickley, D. and Guha, R., *Resource Description Framework (RDF) Schema Specification*. 2000, World Wide Web Consortium. (URL: <http://www.w3.org/TR/2000/CR-rdf-schema-20000327>).
- [2] Budzik, J., Bradshaw, S., Fu, X., *et al.*, *Supporting Online Resource Discovery in the Context of Ongoing Tasks with Proactive Software Assistants*. Intl. Journal of Human-Computer Studies, 2002. **56**(1): p.47-74.
- [3] Carr, L., Bechhofer, S., Goble, C., *et al.* *Conceptual Linking: Ontology-based Open Hypermedia*. In *Proc. of the 10th Intl. WWW Conf.* 2001. Hong-Kong.
- [4] Cunningham, H., Maynard, D., Bontcheva, K., *et al.* *GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications*. In *40th Anniversary Meeting of the Association for Computational Linguistics (ACL)*. 2002. Pennsylvania, US.
- [5] DAML-Coalition, *Reference description of the DAML+OIL ontology mark-up language*. 2001, <http://www.DAML.org/2001/03/reference.html>.

- [6] Domingue, J., Dzbor, M., and Motta, E., *Semantic Layering with Magpie*, In *Handbook on Ontologies in Information Systems*, Staab, S. and Studer, R. (Eds.) 2003, Springer Verlag.
- [7] Domingue, J., Dzbor, M., and Motta, E. *Magpie: Supporting Browsing and Navigation on the Semantic Web*. In *Proc. of the Intelligent User Interfaces (IUI)*. 2004. Portugal.
- [8] Dzbor, M., Domingue, J., and Motta, E. *Magpie: Towards a Semantic Web Browser*. In *Proc. of the 2nd Intl. Semantic Web Conf.* 2003. Florida, USA.
- [9] Eisenstadt, M. and Dzbor, M. *BuddySpace: Enhanced Presence Management for Collaborative Learning, Working, Gaming and Beyond*. In *JabberConf Europe*. 2002. Germany.
- [10] Esenther, A.W. *Instant Co-Browsing: Lightweight Real-time Collaborative Web Browsing*. In *Proc. of the 11th International WWW Conference*. 2002. Hawaii, USA.
- [11] Fensel, D. and Motta, E., *Structured Development of Problem Solving Methods*. IEEE Transactions on Knowledge and Data Engineering, 2001. **13**(6): p.913-932.
- [12] Gruber, T.R., *A Translation approach to Portable Ontology Specifications*. Knowledge Acquisition, 1993. **5**(2): p.199-221.
- [13] Handschuh, S., Staab, S., and Maedche, A. *CREAM - Creating relational metadata with a component-based, ontology driven annotation framework*. In *Proc. of the Intl. Semantic Web Working Symposium (SWWS)*. 2001. California, USA.
- [14] Heflin, J. and Hendler, J., *A Portrait of the Semantic Web in Action*. IEEE Intelligent Systems, 2001. **16**(2): p.54-59.
- [15] Kahan, J., Koivunen, M.-R., Prud'Hommeaux, E., et al. *Annotea: An Open RDF Infrastructure for Shared Web Annotations*. In *10th Intl. WWW Conf.* 2001. Hong-Kong.
- [16] Lieberman, H., Fry, C., and Weitzman, L., *Exploring the web with reconnaissance Agents*. Comm. of the ACM, 2001. **44**(8): p.69-75.
- [17] Middleton, S., DeRoure, D., and Shadbolt, N. *Capturing knowledge of user preferences: Ontologies in recommender systems*. In *Proc. of the ACM K-CAP*. 2001. Victoria, Canada.
- [18] Motta, E., *Reusable Components for Knowledge Modelling*. Frontiers in AI and Applications. 1997, The Netherlands: IOS Press.
- [19] Motta, E., Buckingham Shum, S., and Domingue, J., *Ontology-Driven Document Enrichment: Principles, Tools and Applications*. Intl. Journal of Human-Computer Studies, 2000. **52**(5): p.1071-1109.
- [20] Motta, E., Domingue, J., Cabral, L., et al. *IRS-II: A Framework and Infrastructure for Semantic Web Services*. In *Proc. of the 2nd Intl. Semantic Web Conf.* 2003. Florida, USA.
- [21] Mulholland, P., Zdrahal, Z., Domingue, J., et al., *Integrating working and learning: a document enrichment approach*. Journal of Behaviour and Information Technology, 2000. **19**(3): p.171-180.
- [22] Popov, B., Kiryakov, A., Kirilov, A., et al. *KIM - Semantic Annotation Platform*. In *Proc. of the 2nd Intl. Semantic Web Conf.* 2003. Florida, USA.
- [23] van Harmelen, F., Hendler, J., Horrocks, I., et al., *OWL web ontology language reference*. 2002, <http://www.w3.org/TR/owl-ref/>.
- [24] Vargas-Vera, M., Motta, E., Domingue, J., et al. *MnM: Ontology Driven Semi-automatic and Automatic Support for Semantic Markup*. In *Proc. of the 13th European Knowledge Acquisition Workshop (EKAW)*. 2002. Spain.