MAZETTE: Multi Agent MUSETTE for Sharing and Reusing Ontologies

Jesus Arana, Salima Hassas, and Yannick Prié

Equipe Cognition, Expérience et Agents Situés
LIRIS FRE 2672 CNRS
Université Claude Bernard Lyon 1
Bâtiment Nautibus – 8 Bd Niels Bohr
69622 Villeurbanne
http://experience.univ-lyon1.fr/
{jarana, hassas, yprie}@liris.cnrs.fr

Abstract. During the realization of a document-mediated collective task the participants act and interact by creating documents and ontologies, by modifying them, annotating them and exchanging them. This article presents the general principles of a model based on a multi-agent architecture, aimed at facilitating the co-construction of common ontologies. The model is built on the MUSETTE (Modelling USEs and Tasks for Tracing Experience) approach, which was designed to provide users with assistance based on the recording and reusing of their system use traces. Our model, called MAZETTE (Multi-Agent MUSETTE), defines a framework for considering sharing and reusing of collective traces and experience, amongst which we focus on ontology co-construction and reuse.

1 Introduction

Nowadays, the use of computers is getting not only useful but also essential, as a lot of works are becoming computer-mediated tasks. So from the mixed of the increasing number of persons using computers and the emergence of hundreds or thousand of new softwares and applications, has create the need to develop software agents that act as assistants to aid the users to realise their new tasks.

With the improvement of the information systems, the Internet, and the number of computers growing, the augmentation of web pages submitted by the users has grow in an exponential way. That's one of the main reasons of the existence of the semantic web. One of the purposes of the semantic web , is that computers will be able to use the data on the web for automation, integration and reuse of data across various applications[1] not just for display purposes. To accomplish this, it requires at least two

[1] http://www.w3.org/2001/sw/

R. Meersman et al. (Eds.): OTM Workshops 2004, LNCS 3292, pp. 741–752, 2004.

things, first, data should be structured, this permits computers to distinguish and identify pieces of data; the second reason is that these pieces of data must be described in such a way, that the meaning it can be exploited by machines. This is obtained when the relevant information is represented in a declarative and semantic precise way, and when it is thus understandable for the computer. And this is very important to consider, cause other way, it will not be possible for the computer to assist the user.

Considering all these factors, we are interested by the manipulations of a collective group, when it is interacting in order to realise a computer collective-mediated task. To achieve our analyse we are going to model the users' traces considering a shared knowledge space made up of documents, ontologies and as well as annotations on these documents. In other words, the knowledge space that is available to a user to carry out his tasks. However, this modelling is kind of complex due all the different factors that must be considered, and for that reason, we are going to propose a model based in a multi agent architecture [1], since an agent is an informatics system located in an environment that he can perceive, and that he can act over in an autonomous manner. One of the main objectives of this research is to facilitate the co-construction of ontologies, during the users' interaction.

1.1 Issues in Ontology Sharing and Reuse

It exist several definitions of ontology, the most cited is "*an ontology is an explicit specification of a conceptualisation*" [2], for our context we are going to chose this one: "an ontology provides the common vocabulary of a specific domain and defines, more or less formally, terms meaning and some of their relationships" [3].

So ontologies are extremely important or even essentials to represent the knowledge, because different systems may use different names, even for the same kind of entities; or even worst, they may use the same names for different entities. That's why, the importance of constructing ontologies, to communicate and share knowledge between different users.

In this paper, we propose a model based on a multi-agent architecture, which facilitates the co-construction (by emergence) of common ontologies. This model is based on the MUSETTE approach (Modelling USEs and Tasks for Tracing Experience) [4], a general framework for representing concrete experience in relation with its context of use. So the MAZETTE approach (Multi agent MUSETTE), propose a model based in a multi-agent system to share and reuse collective experience. The approach models the traces of different users in an informatics system. At first, it has a use model that describes all the interest objects such as entities, transitions and relations. After that, the users themselves can track their own use traces, which can be stored in databases. So the purpose of these data is that they can be shared and reused by different users to realise collective tasks.

In the next section, we are going to present the existing related work. In section 3 we present the MUSETTE approach followed by the MAZETTE approach in section 4, where we present the use model, as well as a scenario and we describe the Mazette interface. Finally in section 5 we present our conclusions followed by the references.

2 Related Works

The goal of the semantic web is a web where resources are *machine understandable*, and where the information can be exchanged and processed in an automatic way by persons and agents. So a very important feature of the web ontology language, is that it must have a well defined semantics. OWL (Web Ontology Language) [5], can be used to represent the meaning of terms in vocabularies and their relationships. (This representation of terms and their relationships is an ontology). This language has more capabilities to express meaning and semantic for not only XML, RDF, RDF-Schema since OWL has the ability to represent machine interpretable content on the web.

Related works to achieve the semantic web is through the semantic annotation, and it exists where the content of annotation consists of some more rich semantic information. This idea of Semantic Annotation has been pursued in Ontobroker [6], SHOE [7] and the COHSE project [8]. The objective of Ontobroker is the creation and exploitation of rich semantic structures for machine-supported access to and automated processing of distributed, explicit and implicit knowledge. SHOE allows web page authors to annotate their web documents with machine-readable knowledge. SHOE makes real intelligent agent software on the web possible. In this approach ontologies are mark up in an extension to HTML, and are stored and referred to using an URL. The COHSE project has like objective to improve significantly the quality, consistency and breadth of linking of web documents at retrieval time (browsing documents) and authoring time (as authors creating documents).

With respect to related works in the creation of ontology-based metadata by semantic annotation, we found that these frameworks are some of the most important: CREAM [9, 10, 11], ANNOTEA [12, 13, 14, 15], WebKB [16], and the MnM annotation tool [17]. The CREAM project provides a framework for relational annotation metadata, as well as to annotate HTML pages in order to make their content available to software agents with inference capabilities. ANNOTEA is a tool that enhances collaboration through shared metadata based web annotations. Allows the annotation of web resources with comments. It relies on RDF-Schema, and is a tool that shares the idea of creating a kind of user comments about web pages. WebKB is a Web knowledge-based server based on conceptual graphs to represent the semantic concept of web documents. It embeds conceptual graphs statements into HTML pages. The MnM Annotation tool allows automated and semi-automatic support for annotating web pages with semantic contents. MnM integrates a web browser with an ontology editor and provides open APIs to link to ontology servers and for integrating information extraction tools.

With respect to multi-agents systems research in [18] is proposed a distributed ontology development environment in a multi-agent environment, and a common ontology was build between different users, this approach is convenient when the ontologies of the different parties are stable; but when they are in continuous evolution, is not very good. In the InfoSleuth Project [19], their agents utilise multiple ontologies in order to increase the chances of finding a semantic match of concepts, but not to discover relationships between concepts in the different ontologies. By the other hand the DOGGIE project [20, 21] (Distributed Ontology Gathering Group Integration Environment) proposed how agents with several ontologies can locate and translate semantic concepts distributed among them, in order to share knowledge by automated methods and agent communication strategies. In this approach, they do not assume that ontologies shared commonly labelled concepts, but rather a distributed collective memory of objects that can be categorized into the agent's ontology.

In our model we propose a different approach that take the principles of MUSETTE to track and model the users' manipulations with the objective of reuse the experience. The model is based on a multi agent system and considers users that share a documentary space with the objective of the achievement of a collective task. Our idea is to model each user's traces as a sub-graph. So the total experience is going to be represented like a total graph, where we are going to extract the interesting things, like the relations amongst different users' experience. (Co-construction of emergent ontologies).

3 MUSETTE

The MUSETTE approach is born from the idea of modelling a users' traces with the focus in order to assist him to carry out his tasks (Modelling USEs and Tasks for Tracing Experience) [4] defined at CEXAS (*Cognition Experience et Agents Situés*) at the laboratory LIRIS. The general principles of MUSETTE are described in figure 1. In this approach, there is a user who interacts with a system to modify his documents in his workspace (limited by its files, its software and all that the user can handle). In this system, there is an observant agent that is guided by an observation model that generates primitive traces by respecting a use model starting from these interactions that he observes. Then a trace's generic analyser extracts (starting from

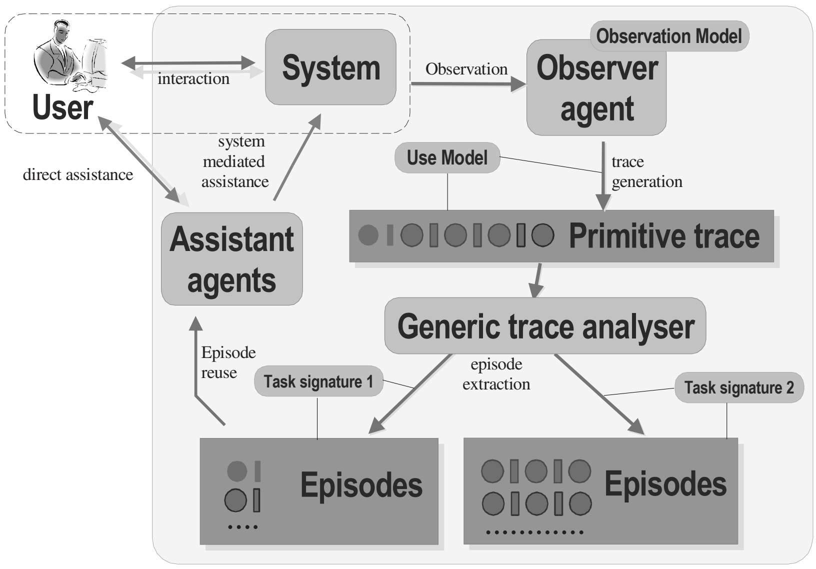


Fig. 1. MUSETTE Approach

the primitive trace) the significant episodes in agreement with signatures from beforehand-defined explained tasks. These episodes will be able to be used by assistant agents, which will be able to assist the user either in a direct way or in an indirect way (instrumentation of the system).

The observation model defines the vocabulary and the necessary rules to determine the relevant data in order to allow the effective construction of the trace, while the use model, describes the objects of interest to be observed which will be registered in the trace, such as the entities (the objects present for the user in its interaction with the system), the events (the objects which occurs during the interaction) and their relations (the binary relations between the entities and the events, entity-entity, entity-event or event-event).

In [4] is proposed a simple example of one use model of a Web navigator (figure 2), in whom the entities correspond to the Web pages (Page), with the hyperlinks (Link), the images (Img) and the user preferences (Cust). The events of this use model are the clicks of the user (Click), the pages that's saves (Sav), the favorite pages which marks (Bm), and the changes of language which it carries out (Lang).

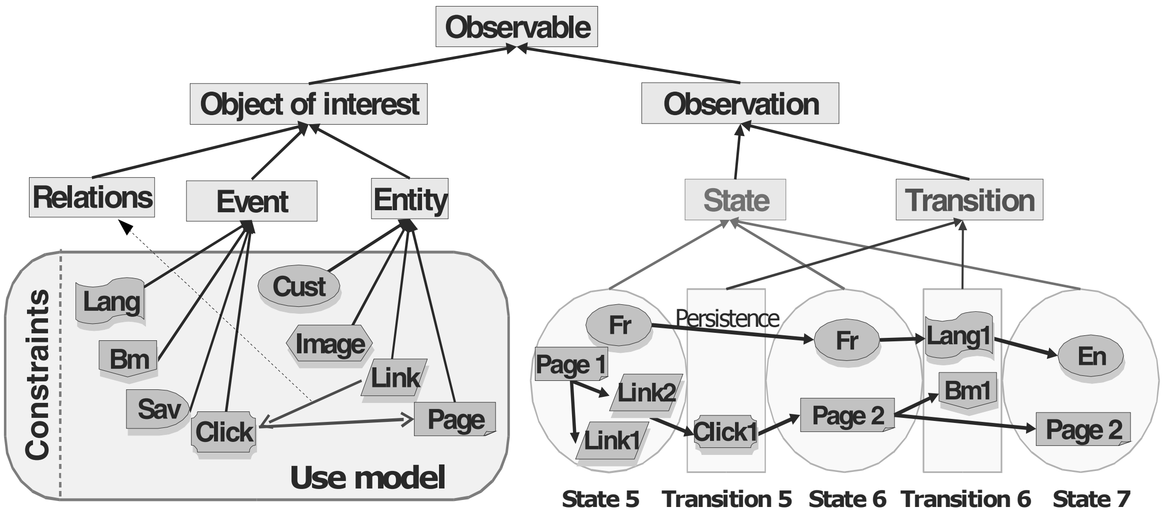


Fig. 2. A simple use model and a web navigation trace

The trace is a representation of the interaction of the user with the system, and it has two structures: states (made up of entities, which represents the state of the system at a given *time*) and transitions (made up of events produced between two states). A trace will be an alternation of states and transitions. One can note in the example, which a change of state is registered in the trace each time that a page is reloaded. An episode is a segment of the trace that represents an experience in the realization of a observed particular task. An episode is determined by a explained task signature (EXTASI) who allows us to find similar experiences, i.e. observations identified for a particular task. The episodes are not cases strictly in the sense of CBR [23], but they are regarded as forming part of bases of potential cases dependent to the EXTASI making it possible to extract them, and as such reusable. In the generation of tasks signatures, there are two characteristics which are opposed: versatility and system effectiveness; in a very general model, there will be different tasks which will not be differentiated the ones from the others, and the episodes will be very vague for beings re-used in an effective way. On the other hand, if the system is very specific, therefore some tasks outside its goal will become undetectable.

4 MAZETTE Approach

4.1 MAZETTE in General

Our approach consists to generalize the MUSETTE approach to assist the user in the process of the sharing and the reuse of the experience during the realization of a collective task. The Mazette approach principles are illustrated in figure 3. First of all, one considers that each user handles a *knowledge space* made up of documents, ontologies, as well as annotations on these documents, with concepts of the ontologies. Knowledge spaces are at disposal of the user within the framework of the realization of its task.

For each user, one considers a software agent *alter ego* which is its representative in the system and which has as task, to provide him its personal ontologies and to assist it according to the experience. This agent alter ego accumulates experience starting from the action of the user in the knowledge space.

To model this experience, we base ourselves on a use model, which we built in accordance with the definitions of the Musette model.

The agent alter ego observes and generates traces according to this use model, those correspond to a representation of the knowledge space and its evolution. These traces will be generated (according to a use model) and stocked in databases, as well as the EXTASIS. This way the alter ego might assist the user by making queries according to the EXTASIS. The alter ego will be able to assist the users starting from episodes segments in the trace according to the explained tasks signatures (EXTASI).
Besides this *traditional* assistance (MUSETTE model applied to an agent alter ego), one considers in MAZETTE (Multi Agent MUSETTE) an assistance in which the alter ego will use the collective experience of the user of more or less shared documentary spaces. It is then a question of considering the total trace left in the general knowledge space, composed of the traces generated by each alter ego. The experience of each user will be modelled as a sub-graph, so the global experience of all users will represent a global graph.

The context that we are considering for our application is for people who will make distance teaching and which will handles documents with many applications. It would be interesting to track the way, in which they work, to design agents that will provide assistance to new users when they are creating a new course, or to improve the way of doing it.

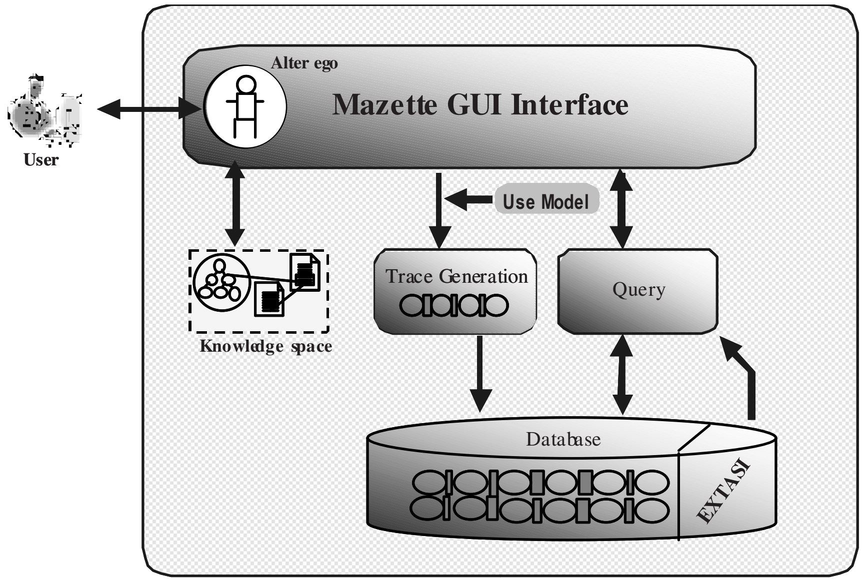


Fig. 3. Mazette Approach

4.2 Use Model

The use model describes the whole of the objects of interest (OIs) that the modeller of the experience wants to analyse, all that he considers important to represent in the trace; i.e. the whole of the entities, the events and the relations. The choice of all these objects of interest is strongly related to the assistance that he wants to provide. Therefore, if one wants to bring an assistance of low level, OIs will be simple and close to the system observed, by the other hand if the desired assistance is of high level, OIs to be tracked will be complex and abstract.

- ***Entities***

We said that the complexity or the simplicity of OIs to be modelled corresponds to the degree of complexity of the assistance to be provided. In our case, where we are in the course of construction of the use model, OIs are limited enough. Initially we have the entity {User} and the entities corresponding to each application in which one will track the actions which occur on these entities: {Word}, {Excel}, {Navigator}. Another level we have the {Documents}, the {Annotations}, the {Ontologies}, etc.

- ***Events***

The events explain us what occurs in the workspace, for example the various actions in which the user utilise a functionality of the system, such as {Create new Doc.}, {Cut}, {Copy}, {Paste}, {Save}, {Annotate}, {Delete}, {Create new concept} in the case of an ontology.

- ***Relations***

The relations are binary and oriented, and they can exist as much as for the entities that for the events, i.e. relations between entity-entity, event-event or entity-event, the number of the relations will be according to the level of the desired assistance. For example, a relation of very low level would be when a piece of the text is selected, and will not be relevant only when the user selects it to be copied and pasted it in another document.

4.3 Scenario

To visualise the things that we can do, with our model, let's imagine a scenario where three users making some manipulations, the systems is going to observe and this way to stock the traces in order to save these manipulations.

User 1. The user execute the Internet application querying the web to search web pages for "Web Semantic" & "RDF". After he found the results, the user click over a link named "Semantic web", and finally he bookmark this site.
User 2. The user annotates a file named "The professions" with a domain ontology. After that he e-mail the file to a contact.
User 3. The user annotates a web page, in an annotation server. By the other hand, he annotates a web page of Industrial Engineering, with a domain Ontology.

In the figure 4 we show all these manipulations created with a tool to draw a user's trace with respect to a use model; this tool is a plug-in for Protégé 2000 realised at the laboratory LIRIS [24].

Once the users have use their systems, the model is going to be more intelligent, in others words, after we use the system, this one, it will learn, from the preferences of the user as well as the way it works.

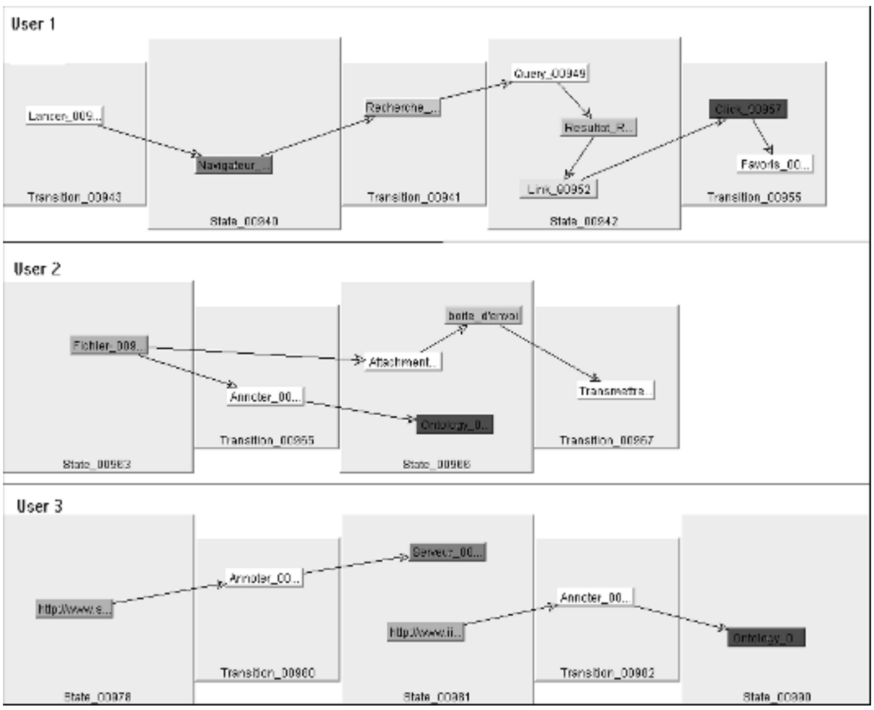
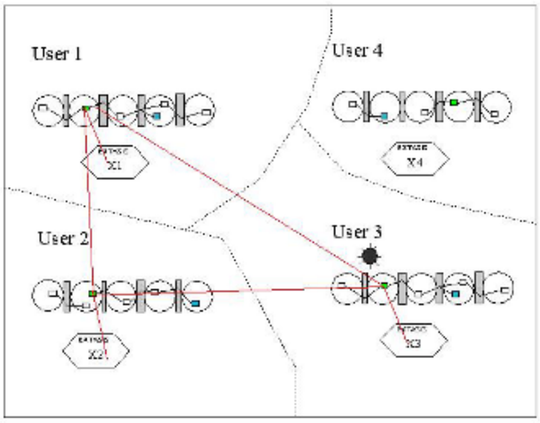


Fig. 4. Use scenarios.



	EXTASIS 1
	Create a document and annotate it with a concept of an ontology
User 1	Create a document word and annotate it with the concept Annotations of the Computer Systems Ontology
User 2	Create a document powerpoint and annotate it with the concept Annotations of the Computer Systems Ontology
User 3	Create a document word and annotate it with the concept QCM of the Computer Systems Ontology

Fig. 5. Sub-graph.

The way where we will store the use's traces, the task signatures and the extasis will be through databases of the MySQL type. The tables will be all the various entities, the events and their relations. In this way, one will be able to store all handling made by the users and they can be retrieved in the same way.

Agents are going to be moving from the database of each and all the users that share the system, and their focus is to find interesting things. For example for the first user that has bookmarked one page, one possible assistance may be to find similar web sites, or find who else has bookmarked a specific site. For the second scenario, that the user has bookmarked a file with a domain ontology, one possible assistance may be, to find what else has been bookmarked with the same ontology. For the third scenario, that the user has annotate a web site in an annotation server, and a web site with an ontology, it will be interesting of read the annotations that other people has made from some web pages, when we are navigating those pages. An example of how the traces can be seen like a sub-graph is shown in figure 5.

4.4 Mazette GUI, the Application

Initially, being given there is not a tool to manage the annotations on all the types of documents that we would like to treat, we will create a graphic interface which will be used to seize the traces according to a Musette ontology by one side, and following personal ontologies to the user by the other side, which will be useful to manage the ontologies of the user, as well as his annotations. Currently, we are working developing this application in Java due its portability, amongst several advantages.

In this graphic interface the user:

- will build use traces of him-self (because for the moment we cannot develop a tool for automatic tracing); who will be stored directly into a MySQL database.
- will manage its different ontologies;

- will annotate the documents by using concepts of its ontologies.
- Will query the database.

From the trace will be extracted the episodes following the task signatures, which could be used to assist the user (in a more or less complex way).

Once the chance of building individual traces in the alter ego provided, we will model the total experience of the traces of several alter egos in the shape of a total graph. We will be able to determine in this way one of the interesting forms corresponding to collective tasks (generalization of task signatures), but to also make emerge, according to the interactions agents, new knowledge of assistance (for example interesting links). The total system will be implemented in a FIPA architecture by using the platform JADE (Java Agent DEvelopment Framework).

5 Conclusion

This project is in the convergence of the domains of SMA [1], the task modelling [25], and ontologies [2] (since an ontology is used to express a knowledge base in a formal language of representation, and such a way that a computer can use it).

In this paper, we presented the general principles of MAZETTE model. This project was born from the need of modelling the actions of a user while they are interacting with the objective of a collective computer-mediated task in order to build intelligent systems that are able to provide him assistance. The goal of the MAZETTE project is to concretise in one application that will be able to store and track the user's manipulations, will manage its different ontologies and will be able to annotate his documents through his ontologies. All this experience will be represented as a partial graph, and the graph global will be the addition of all users' experience, so they can share it. Future works includes the creation of a tool for automatic use tracing, and the final objective is to achieve into a web tool for sharing and reusing experience.

There are work which proceeds within the team in the same direction and are based on the Musette model; like, the modelling of the experience in a cartographic software directed towards the *veille informatique* in [26], and the development of a MemSim prototype to share the experience in a collaborative activity of integrated design [27].

References

[1] Sycara, K.P., 1998. Multi agents Systems. AI Magazine, Vol 10, No. 2, 1998, pp. 79-93.

[2] Gruber, T. R., 1993. A translation approach to portable ontology specifications. Knowledge Acquisition, 5(2), 1993.

[3] A. Gomez-Perez. Développements récents en matière de conception, de maintenance et d'utilisation des ontologies. In Proceedings of colloque Terminologie et intelligence artificielle de Nantes, 10-11 mai 1999, revue terminologies nouvelles, pp. 9-20.

[4] Champin P.-A., Prié Y., Mille A., 2003. MUSETTE: Modelling USEs and Tasks for Tracing Experience ,Proc. From structured cases to unstructured problem solving episodes - WS 5 of ICCBR'03, Trondheim (NO), NTNU, Trondheim (NO), pp279-286, June 2003. http://www710.univ-lyon1.fr/~champin/publis/iccbr2003b.pdf
[5] OWL . http://www.w3.org/TR/owl-features/
[6] Decker S., Erdmann M., Fensel D., Studel R., 1999. Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In R. Meersman, Z. Tari,, and S. Steevens, editors, Semantic Issues in Multimedia Systems. Proceedings of DS-8, pages 351-369. Kluwer Academic Publishers.
[7] Heflin Hendler J., Luke S., 1999. SHOE: A Knowledge Representation Language for Internet Applications. Technical Report CS-TR-4078 (UMIACS TR-99-71), Department of Computer Science, University of Maryland.
[8] Carr L., Bechhofer S., Goble C. A., Hall W., 2001. Conceptual Linking: Ontology-based Open Hypermedia. In Proceedings of WWW10, Tenth World Wide Web Conference, Hong Kong, May 2001.
[9] Handschuh S., Staab S., Maedche A., 2001. CREAM – Creating relational metadata with a component-based, ontology-driven annotation framework. In proceedings of K-CAP 2001, First International Conference on Knowledge Capture, Victoria, B.C. Canada, October 2001.
[10] Handschuh S., Staab S., Maedche A., 2002. CREAting relational Metadata (CREAM) – a framekork for semantic annotation. Frontiers in artificial intelligence and applications. IOS press, Amsterdan (NL).
[11] Handschuh S., Staab S., Maedche A., 2003. Annotation of the shallow and the deep Web. Annotation for the Semantic Web, eds. Siegfried Handschuh and Steffen Staab. IOS Press, Frontiers in Artifical Intelligence and Applications. Vol 96. 2003.
[12] Kahan J., Koivunen M.-R., Prud'Hommeaux E., Swick R., 2001. Annotea: An Open RDF Infrastructure for Shared Web Annotations. In Proceedings of the Tenth International World Wide Web Conference, Hong Kong , May 2001.
[13] World Wide Web Consortium. Annotea Project. http://www.w3.org/2001/Annotea.
[14] Koivunen M.-R., Brickley D., Kahan J., Prud'Hommeaux E., and Swick R., 2000. The W3C Collaborative Web Annotation Project … or how to have fun while building an RDF infrastructure.
http://www.w3.org/2000/02/collaboration/annotation/papers/annotationinfrastructure, May 2000.
[15] Koivunen M.-R., Swick R., 2003. Collaboration through Annotations in the Semantic. Annotation for the Semantic Web, eds. Siegfried Handschuh and Steffen Staab. IOS Press, Frontiers in Artifical Intelligence and Applications. Vol 96. 2003.
[16] Martin P. and Eklund P., 1999. Embedding Knowledge in Web Documents. In Proceedings of the 8th Int. World Wide Web Conf. (WWW'8), Toronto, May 1999, pages 1403-1419. Elsevier Science B.V.
[17] Vargas-Vera M., Motta E., Domingue J., Lanzoni M., Stutt A., Ciravegna F., 2002. MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup. In EKAW02, 13th International Conference on Knowledge Engineering and Knowledge Management, LNCS/LNAI 2473, pages 379-391. Siguenza, Spain, October 2002. Springer.
[18] Takaai, M., H. Takeda, and T. Nishida (1997). Distributed Ontology Development Environment for Multi-Agent Systems, Working Notes for AAAI-97 Spring Symp. Series on Ontological Engineering, pp. 149-153.

[19] Bayardo R., Bohrer W., Brice R., Cichocki A., Fowler J., Helal A., Kashyap V., Ksiezyk T., Martin G., Nodine M., Rashid M., Rusinkiewicz M., Shea R., Unnikrishnan C., Unruh A., Woelk D., 1998. InfoSleuth : Agent-Based Semantic Integration of Information in Open and Dynamic Environments, 205-216, Morgan Kaufmann, San Francisco, 1998.

[20] DOGGIE: A Multiagent System for Learning Diverse Web Ontologies, Andrew B. Williams, Costas Tsatsoulis; Seventeenth National Conference on Artificial Intelligence; Austin, TX, July 2000.

[21] Williams, B. A. and Z. Ren (2001). Agents Teaching Agents to Share Meaning, Proc. ICMAS2001, ACM Press, Montreal, Canada, 465-472.

[22] http://experience.univ-lyon1.fr/

[23] Fuchs B., Mille A., 1999. Une modélisation au niveau connaissances de la tâche d'adaptation en raisonnement à partir de cas. In Actes de RàPC'99, Palaiseau, Juin 1999 , pages 27-36., AFIA, 1999.

[24] Huvier E., 2003. Rapport de Travail de Fin d’Etude Atelier de maquettage pour la modélisation de l ‘expérience d’utilisation d’une application informatique. LIRIS, Sep 2003.

[25] Schreiber G., Hakkermans A., Anjewierden R. de Hoog, Shadbolt N., Van de Welde W et Wielinga B, 1999. Knowledge Engineering and Management : The CommonKADS methodology. The MIT Press, 1999.

[26] Laflaquière J, 2003. L’expérience d’utilisation d’un espace documentaire pour améliorer l’assistance à l’exploitation de l’information dans le cadre de la veille. Mémoire de DEA informatique, Université Lyon 1.

[27] Stuber A., Hassas S., Mille A., 2003. Combining MAS and Experience Reuse for assisting collective task achievement, in proceedings of CBR-2003 Workshop « From structured cases to unstructured problem solving episodes for experience-based assistance », Trondheim, Norway, June 23 to 26, 2003, Lecture Notes in Computer Science-Springer Verlag.