

A RULE-BASED APPROACH FOR CUSTOMIZING KNOWLEDGE SEARCH IN CNOS

Rui J. Tramontin Jr. ¹, Chihab Hanachi ², Ricardo J. Rabelo ¹

¹ Federal University of Santa Catarina, Department of Automation and Systems
GSIGMA – Intelligent Manufacturing Systems Group, Florianópolis (SC), BRAZIL
tramontin@gsigma.ufsc.br
rabelo@das.ufsc.br

² Institut de Recherche en Informatique de Toulouse IRIT, UMR 5505, Université
Toulouse1, 2 rue du Doyen-Gabriel-Marty, 31042 Toulouse Cedex 9., FRANCE
hanachi@univ-tlse1.fr

Searching for knowledge in Collaborative Networked Organizations (CNOs) is an important issue as partners must share and use the knowledge spread over the network. Besides that, partners of such networks work in several contexts (roles, activities, processes) and have individual interests. Based on these observations, the aim of this work is to provide and combine concepts such as topics, profiles and context in a model for customizing knowledge search in CNOs. The basic assumption is that the relevance of the search results in the CNO domain is not only defined by the terms of the query but also by the context and the profile of the user performing the search. Besides the model, a set of rules for query customization is presented and all these elements are framed in an existing framework for knowledge search.

1. INTRODUCTION

Due to new characteristics of the global economy and society, new forms of inter-organizational interactions are emerging and have been classified into the general concept of Collaborative Networked Organization (CNO). A CNO can be defined as group of distributed, heterogeneous and autonomous entities (organizations or people) that formally collaborate in response to business opportunities and their interactions are supported by computer networks (Camarinha-Matos, 2005).

As collaboration among CNO partners is of paramount importance, the quality of the information / knowledge exchanged and searched may have a relevant impact on the performance of the network. In particular, the search for CNO knowledge is essential for supporting the different activities during the CNO life cycle like partners selection, performance indicators selection, *CNO management* and *inheritance*.

Knowledge search rises up as an alternative in order to support these and other activities that are performed by partners playing some role in the CNO, like managers, brokers, planners, or even simple members. Knowledge search in CNOs can be represented by the abstract architecture shown in Figure 1. In the top layer, there are *knowledge consumers*, which are the actors that perform knowledge search: CNO applications in general and CSCW tools. The second layer presents a

knowledge search system, which is abstracted by a search engine and its index repository. The CNO knowledge is located in the third layer, and it is defined as the combination of concrete information and the CNO ontology (linked by means of semantic annotations). Concrete information is stored in documents and databases such as: a common CNO repository, (contracts, business process specifications, templates), legacy databases, forums, wikis, weblogs, etc (resulting from interactions among partners). CNO knowledge is also composed of additional general-purpose metadata (authors, title, date, etc.) and comments from CNO partners (different views / opinions). Finally, the bottom layer is composed by *knowledge providers*, which are the applications that make CNO knowledge available. It is important to highlight that in some cases applications are simultaneously knowledge providers and consumers.

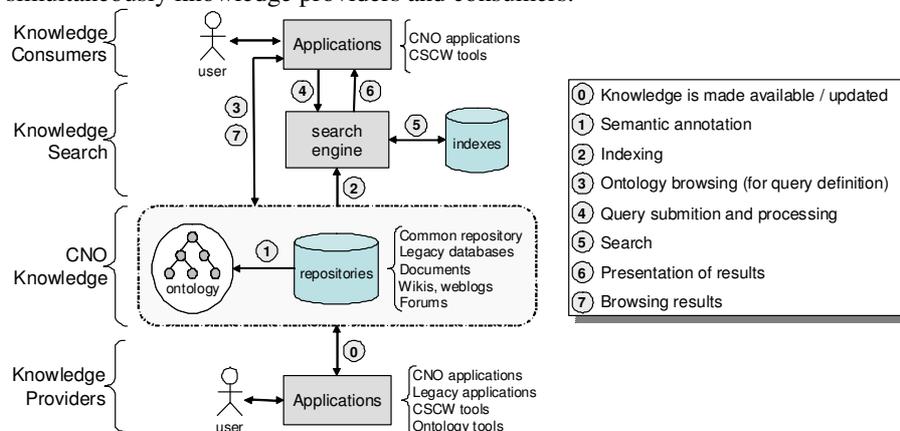


Figure 1 – Knowledge Search in CNOs.

This picture also shows the main supporting functionalities for knowledge search. Each supporting functionality can be seen as a sub-problem: 1) *semantic annotations* can be defined according to several techniques related to information extraction and natural language processing (Uren, 2006); 2) *indexing* of the knowledge (it means, semantically annotated information) should take into account that such knowledge is distributed over the CNO, so a strategy for locating it should be defined (for instance, using a *crawler*); 3) *ontology browsing* requires easy-to-use interfaces (usability problem); *query definition* assumes the existence of a language for representing queries; 4) *query processing* involves the expansion and / or customization of the query (Bhogal, 2007); 5) *search* involves how to define a relevance order for the results according to the query; 6/7) *presenting and browsing results* are also related to usability.

This paper focuses on the functionality for query processing, particularly in *query customization*. Under this perspective, *customization* means to take the *context* into account in order to adapt information and services in a specific way to match the unique and specific needs of an individual user or a community of users (Renda, 2005). This subject is a well-explored theme in the literature, being also referred as *personalization* or *adaptation*. It is used in a wide range of application domains such as digital libraries (Renda, 2005), information gathering (Bouslimi, 2008), and scheduling in CNOs (Almeida, 2007).

The customization of the knowledge search in the scope of a CNO is a very relevant problem as the user’s context has an impact on the knowledge search and access. Such context is defined by the CNO where each user is taking part, his/her role (or the role of the organization he/she works for), the activity and process he/she is involved in, the life cycle stage. Another aspect to be considered is the individual interests of the user, that is, the profile. The basic assumption is that the relevance of the search results in the CNO domain is not only defined by the terms of the query but also by the context and the profile of the user performing the search. In this sense, the same query provided by users (or the same user) in different contexts will provide different (customized) results. The benefit of a personalized search is decreasing the time it takes people to find information (Pitkow, 2002).

In order to do customization, four main issues should be addressed: (i) How to represent the context and the user interests (models)? (ii) How to build and maintain the user’s profile (search history / collaborative filtering)? (iii) How to capture the context (it may vary according to the nature of the context)? (iv) How to perform the personalization (query expansion / recommendation systems)?

This paper focuses on issues (i) and (iv) and proposes both a model and rules for customizing queries using the notion of *context* (dynamic situation), *user profile* (individual interests of the user) and the use of *topics* for structuring both the CNO ontology and the profiles. The work described in this paper is an extension of the architecture for *knowledge search services* proposed by Tramontin Jr. (2007).

The paper is organized as follows. Section 2 describes a model that supports the customization process. Section 3 shows the query customization process through rules. An architecture for extending the current knowledge search services is presented in section 4. Section 5 discusses the related work. Final remarks and next steps of this work are presented in section 6.

2. MODEL FOR CUSTOMIZING QUERIES

In order to allow a system to capture the user’s context and use it for customizing queries, such context should be modeled. Besides that, the user’s interests (profile) should also be represented. Figure 2 shows an ontology that models these and other correlated elements of a knowledge search system: user, query and the CNO ontology.

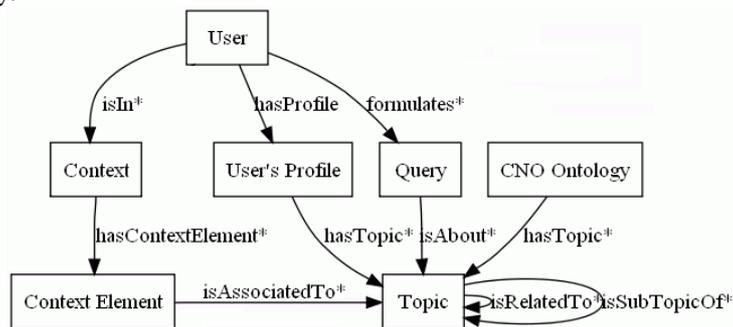


Figure 2 – Ontology for supporting the customization of K. search (with Protégé).

According to this ontology, a *user* has a *profile*, is in a certain *context*, and can formulate *queries*. A *context* is composed by *context elements*, which represent particular aspects of the context, such as the role of the user, the activity and process he/she is involved in, and so forth. *Topics* extend the expressiveness of ontologies (in particular the CNO ontology) by grouping classes and attributes that belong to the same semantic subject. Topics are used both by *profiles* (representing the interests of the user) and by *queries* (referring to the topics the user wants to search).

2.1 User's Profile

As previously mentioned, the *user's profile* is composed by a set of topics to which the user is interested in. It is used to either restrict or provide additional results (not explicitly mentioned in the query) to the user according to his/her topics of interest, and hence providing a customized search.

A *topic* (Bouslimi, 2008) is a view on an ontology and corresponds to a set of classes, attributes and links between them, all being related to the same semantic subject. So, similar to a view in a database, it can filter a set of classes and the attributes of each class belonging to this topic. Nevertheless, unlike a database view, a topic can have subtopics and be related to other topics, allowing the navigation through correlated topics, allowing the user to control the dimension of the *knowledge search space*. Topics provide means to structure the universe of discourse in manageable conceptual units that have a concrete meaning for the user providing a good trade-off between querying at a very low level (attributes) and vague querying at the ontology level.

Examples of topics in a CNO ontology can be: the *organizational* aspect of CNOs (partners, roles, relationships, competencies); the CNO *resources* (hardware, software, people); *management* aspects (control, supervision); *operational* or *functional* issues (tasks, processes), among others. Since a CNO ontology will provide the structure for queries, topics can help on two aspects: facilitating the query definition process and providing additional results.

Topics facilitate the query building process in the sense that the user does not need to define a very detailed query (that would oblige him/her to be aware of the whole ontology schema), but rather to select one or more topics of interest. For instance, instead of defining a query for searching all purchasing orders from a given partner, the user could simply search for the topic "purchasing". Following the same example, the system could suggest additional results concerning the topic "delivery", as it is somehow related to "purchasing".

2.2 Context

Like profile, the *context* is another element that supports query customization. According to Dey (2001), a context is "*any information that can be used to characterize the situation of an entity that is considered relevant to the interaction between a user and an application*". Taking this definition into account, the profile could be considered as part of the context. However, we decided to include in the context only the dynamic aspects (mainly CNO-related issues) and to include the individual interests of users in the profile (although it can evolve over time, it is more static than the information contained in the context).

The context is composed by *context elements*, which in turn can be specialized in *physical* (regarding the geographical location of the user, application and device he is using) and *CNO-related* (the CNO the user is involved in, roles, the activities and processes he is working on, the stage of the CNO life cycle). All these concepts are presented in the fragment of the customization ontology shown in Figure 3.

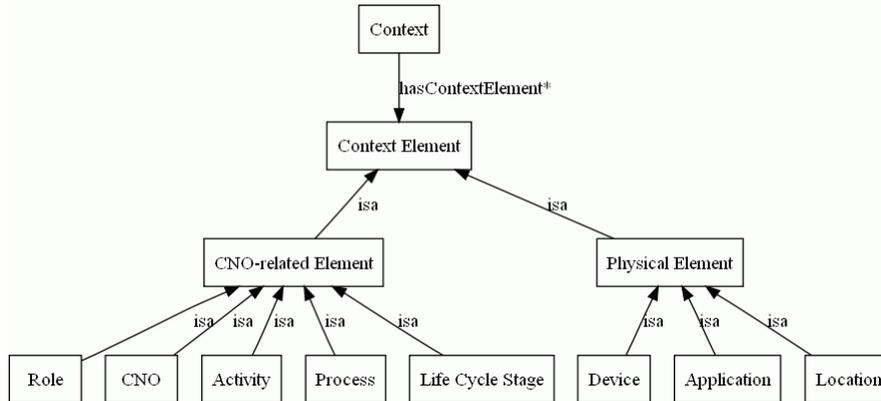


Figure 3 – Elements that compose a context (with Protégé). [Adapted from Kirsch-Pinheiro (2004)]

3. RULES FOR QUERY CUSTOMIZATION

This section illustrates how the model presented in the previous section is used to customize queries. In order to show how this process works, the elements of this model and their relationships will be represented as predicates, and then rules for customization will be presented. Table 1 below shows such predicates.

Table 1 – Customization Model defined as predicates.

	Predicates	Description
user & query	hasProfile (U, UP)	User U has profile UP.
	formulatesQuery (U, Q)	User U formulates query Q.
	isAbout (Q, T)	Query Q is about topic T.
	isIn (U, C)	User U is in context C.
topics & profile	isSubTopicOf (T ₁ , T ₂)	Topic T ₁ is subtopic of topic T ₂ .
	isRelatedTo (T ₁ , T ₂)	Topic T ₁ is related to T ₂ . (symmetric)
	containsTopic (UP, T)	User profile UP contains topic T.
context	hasContextElement (C, E)	Context C has context element E.
	isAssociatedTo (E, T)	Context element E is associated to topic T.

The first three predicates show the relationships between users, profiles and queries. Moreover, there are some predicates for topics: one representing the notion of subtopics (inheritance) and the other one representing the idea of correlated topics (meaning that they have some semantic connections between them). It is important to notice that the *isRelatedTo* predicate is symmetric, which means that if T₁ is related to topic T₂ then T₂ is related to topic T₁. There is another predicate that specifies a profile containing topics. Finally, there are predicates representing

relationships between contexts and context elements and between contexts elements and topics.

3.1 The Customization Process

The query customization process is defined by five rules (declared in Prolog syntax). The first rule adds correlated topics to the ones referred in the query:

$$\text{isAboutRelatedTopic}(Q, T_1) \text{ :- } \begin{array}{l} \text{isAbout}(Q, T_2), \\ \text{isRelatedTo}(T_1, T_2), \\ \text{not}(\text{isAbout}(Q, T_1)). \end{array} \quad (1)$$

Rule 1 says that a query Q is about a related topic T_1 if Q refers to a certain topic T_2 which in turn is related to T_1 , and T_1 is not referred by Q . The purpose of this rule is to add topics that are correlated to the ones mentioned in a query, but not considering either the profile or the context.

In order to extend a query with topics contained in the user's profile, it is necessary to deduce the profile which is associated to this query:

$$\text{isAssociatedToProfile}(Q, UP) \text{ :- } \begin{array}{l} \text{formulatesQuery}(U, Q), \\ \text{hasProfile}(U, UP). \end{array} \quad (2)$$

Based on rule 2, a query Q is associated to a profile UP if user U formulates the query Q and U has UP as profile. Now, having rules 1 and 2, it is possible to customize the query by taking into account the correlated topics contained in the user's profile:

$$\text{isAboutTopicInProfile}(Q, T_1) \text{ :- } \begin{array}{l} \text{isAboutRelatedTopic}(Q, T_1), \quad (1) \\ \text{isAssociatedToProfile}(Q, UP), \quad (2) \\ \text{containsTopic}(UP, T_1). \end{array} \quad (3)$$

Rule 3 says that a query Q is about a topic T_1 in profile if T_1 is related to a topic T_2 (based on rule 1), Q is associated with profile UP (rule 2) and the profile UP contains T_1 . Rule 3 is a more restrict case of rule 1 in the sense that it selects only correlated topics that are also in the user's profile.

Rule 4 deduces the context in which the query was formulated, by taking the context of the user who formulated the query:

$$\text{isFormulatedIn}(Q, C) \text{ :- } \begin{array}{l} \text{formulatesQuery}(U, Q), \\ \text{isIn}(U, C). \end{array} \quad (4)$$

Finally, rule 5 uses rule 4 to select topics associated to the context:

$$\text{isAboutTopicAssociatedToContext}(Q, T_1) \text{ :- } \begin{array}{l} \text{isFormulatedIn}(Q, C), \quad (4) \\ \text{hasContextElement}(C, E), \\ \text{isAssociatedTo}(E, T_1), \\ \text{not}(\text{isAbout}(Q, T_1)). \end{array} \quad (5)$$

Based on rule 5, a query Q is about a topic T_1 associated to the context if the context C in which Q was formulated contains a context element E associated to T_1 and T_1 is not referred by Q .

It is important to mention that the customization is performed by rules 1 (correlated topics), 3 (topics in profile) and 5 (topics associated to the context).

Rules 2 and 4 are supporting rules for the others, responsible for deducing respectively the user’s profile and context.

The use of these rules should be configurable and it is also desirable to allow the user to have control over the customization, by accepting or changing the effects of them. Therefore, once the customized query is defined and accepted by the user, it is submitted to the search engine, which in turn should be able to distinguish the original topics (mentioned in the original query) and the topics deduced in the customization. The purpose of this approach is to allow the system to make a distinction between “normal” results and the “customized” results. It means that although the customization is made in the query, it has impact in the other parts of the knowledge search process (search and presentation of results).

3.2 Example

An example of the customization rules is presented in Figure 4. It is about a VBE¹ scenario, where a user (playing the role of *VO Broker*) submits a query associated to topics of the VBE ontology. The left part of the figure shows an instance of the model containing a user (John) who has a profile (this profile is associated to the topics *Resources*, *Products* and *VO Planning*). The user is taking part of a context (relation “is in”) Context_VO1, as a VO broker. The right side of the figure shows the results of the five rules. In order to facilitate the understanding, this example shows only one context. However, users can be associated to several contexts, and one of them will be captured at runtime in order to support the customization. For instance, this user can be the VBE manager in a different context.

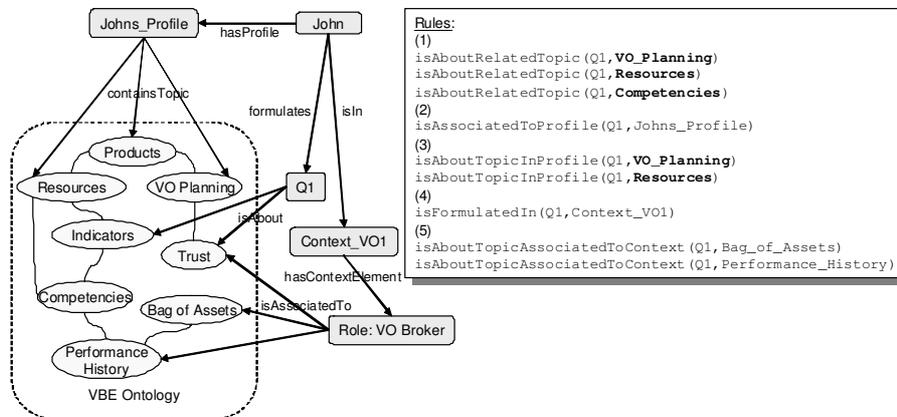


Figure 4 – Example of customization.

According to this example, John formulates a query (Q1) about topics *Indicators* and *Trust*. Rule 1 adds the topics correlated to the ones referred by the query: *Resources*, *VO Planning* and *Competencies*. Rule 3 is more restrictive than 1 as it adds only the correlated topics that are also in the user’s profile (*Resources* and *VO Planning*). Rule 5 adds topics according to the elements of the context. In this case,

¹ Virtual Organization Breeding Environment: this kind of CNO provides support for the formation of dynamic Virtual Organizations (VOs).

topics *Bag of Assets* and *Performance History* are added as they are associated to the user's role (VO Broker). Therefore, considering that the user can accept or not the new topics, Q1 can be extended by adding any subset of {*Resources*, *VO Planning*, *Competencies*, *Bag of Assets*, *Performance History*}. As mentioned, rules 2 and 4 are just supporting rules, responsible to associate the query respectively to the profile (used by rule 3) and to the context (rule 5).

This example considers only the role as contextual element having impact on the customization, but it is not difficult to think on similar scenarios related to users working in a given activity / process, or a combination of role and activity.

4. SYSTEM ARCHITECTURE

This work is an extension of an existing framework for knowledge search in CNOs called *Knowledge Search Services* (Tramontin Jr., 2007). Such framework provides functionalities for ontology management, automatic semantic annotation of documents and ontology-based search. The objective is to add new elements for implementing the model and rules presented in this work as well as to extend the ontology used by this framework with the notion of topics.

From the architectural point of view, the customization is performed by a new component called *query customizer*. The architecture for customized knowledge search can be seen in Figure 5. The new component as well as the repositories for contexts, profiles and rules are highlighted by a dashed rectangle.

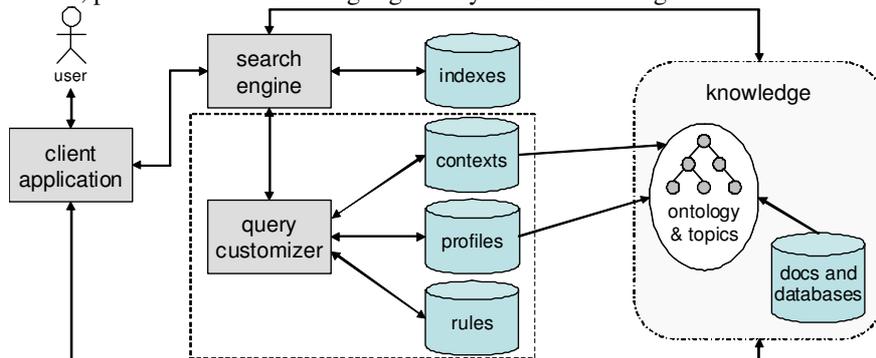


Figure 5 – Architecture for customized knowledge search.

In this architecture, when the client application submits a query, the search engine sends it to the query customizer. The customizer then applies the rules and extends the query according to the user's profile and the current context. The search engine gets the customized query and sends it back to the user. The new query is accepted or changed by the user and it is submitted again to the search engine, which finally performs the search.

5. RELATED WORK

This work shares similar ideas from some correlated areas, namely: context-aware systems, rule-based personalization and query expansion.

Firstly, this work is related to *context-aware systems* as it defines a model for representing the context as well as the user's profile. Specifically, an ontology-based model is proposed, which was adapted from the object-oriented model for context defined by Kirsch-Pinheiro (2004). Other works in this area include Baldauf (2007), who evaluated the existing context-aware systems and models, and Gensel (2008), who presents an approach for adapting information systems in the Web.

Another correlated subject concerns rule-based personalization. *Rules* are used as specific knowledge extracted from historical information (data mining), like done by Mobasher (2001). Such approach of gathering the user's search history is related to *recommendation systems* (Balabanovic, 1997; Renda, 2005) and *information filtering* (Belkin, 1992), and can be seen as a complementary aspect to the focus of this paper. Rules are used in this paper as a more generic and formal way to add correlated topics to queries based on the user's profile and context.

Concerning *query expansion*, the approach used in this paper borrows the same idea, but with a different motivation. Query expansion is the process of adding new terms to the query in order to reduce its ambiguity (Bhogal, 2007), and hence to improve the search efficiency. In this paper, the query is expanded, but the new terms (in this paper, *topics*) are related to the context and profile of the user. In a similar way as the usage of rules, query expansion based on ambiguity can also be seen also as a complementary aspect to the expansion based on the context and profile. Nevertheless, it is important to define ways to compare the performance between these two approaches, and we intend to define measures for doing so in the future.

It is also important to mention that the principle of using both profile and context for query expansion is shared by Pitkow (2002). The difference is that their focus is on Web search, while this paper focuses on customizing knowledge search in the CNO domain.

6. FINAL REMARKS

This paper presented an ontology and rules for customizing knowledge search in CNOs. The objective is to improve the relevance of searches by considering the user's *context* (specially the CNO context: role, activity, life cycle, etc.) and *profile* (individual topics of interest). The approach also used the concept of *topics* (Bouslimi, 2008), which extend the expressiveness of ontologies and thus facilitates the query definition.

An aspect not covered in this paper is how the context is captured. Nevertheless, assuming the existence of a security framework (like the one developed by Sowa (2007)), part of the context (CNO and role) can be obtained when the user logs in the system (authentication).

Next steps involve extending the current implementation of the knowledge search services by adding a new service for query customization as well as the validation of this implementation. The rules will be first tested in Protégé² and then they will be added to the concrete implementation of the customizer. Finally one issue that will be faced is that the notion of topics, though it already exists, it is not implemented by the current ontology languages (for instance, OWL). So, in order to

² <http://protege.stanford.edu/index.html>

be used in practice, such languages must be extended as well as the corresponding implementations.

6.1 Acknowledgments

This work has been partially supported by the Brazilian council of research and scientific development – CNPq. It has been developed in the scope of the Brazilian IFM project (www.ifm.org.br) and the European IST FP-6 IP ECOLEAD project (www.ecolead.org). Special thanks also to Mr. Wassim Bouaziz for his help on the model and rules definition, and Mr. Leandro Loss and Mr. Fabiano Baldo for their valuable comments.

7. REFERENCES

1. Abdelali, A.; Cowie, J.; Soliman, H. S. Improving query precision using semantic expansion. *Information Processing and Management* 43, pp. 705–716, 2007.
2. Almeida, A.; Marreiros, G.; Martins, C. Collaboration and Adaptation in Scheduling. In: *Proceedings of PRO-VE'2007 – 8th IFIP Working Conference on Virtual Enterprises*, 2007.
3. Balabanovic, M.; Shoham, Y. Fab: content-based, collaborative recommendation. *Communications of the ACM CACM* 40 (3), pp. 66-72, 1997.
4. Belkin, N. J.; Croft, W. B. Information filtering and information retrieval: two sides of the same coin? *Communications of the ACM* 35 (12), pp. 29-38, 1992.
5. Baldauf, M.; Dustdar, S.; Rosenberg, F. A survey on context-aware systems. *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. 2, No. 4, 2007.
6. Bhogal, J., Macfarlane, A.; Smith, P. A review of ontology based query expansion. *Information Processing and Management* 43, pp. 866–886, 2007.
7. Bouslimi, I.; Hanachi, C.; Tout, H.; Ghedira K. "Coordination Framework for Cooperative Information Gathering", *International Journal of Advanced Intelligence Paradigms*, Inderscience, to be published 2008.
8. Camarinha-Matos, L. M.; Afsarmanesh; H.; Ollus, M. ECOLEAD: A Holistic Approach to Creation and Management of Dynamic Virtual Organizations. In: *Proceedings of the Sixth IFIP Working Conference on Virtual Enterprises (PRO-VE'05)*. pp. 3-16, 2005.
9. Dey, A. K. Understanding and using context. *Personal and Ubiquitous Computing*, Vol. 5, No. 1, pp 4-7, 2001.
10. Gensel, J.; Villanova-Oliver, M.; Kirsch-Pinheiro, M. Modèles de contexte pour l'adaptation à l'utilisateur dans des Systèmes d'Information Web collaboratifs. In: *Workshop from "8èmes journées francophones"*. Sophia-Antipolis, France. pp. 5-16, 2008.
11. Kirsch-Pinheiro, M.; Gensel, J.; Martin, H. Representing Context for an Adaptive Awareness Mechanism. *CRIWG 2004*: 339-348, 2004.
12. Mobasher, B.; Dai, H.; Luo, T.; Nakagawa, M. Effective Personalization Based on Association Rule Discovery from Web Usage Data. *Proceedings of the ACM Workshop on Web Information and Data Management (WIDM01)*. Held at CIKM 2001, Atlanta, Georgia, November 2001.
13. Pitkow, J. E.; Schütze, H.; Cass, T. A.; Cooley, R.; Turnbull, D.; Edmonds, A.; Adar, E.; Breuel, T. M.. Personalized search. *Commun. ACM* 45(9): pp. 50-55, 2002.
14. Renda, M. A.; Straccia, U. A personalized collaborative Digital Library environment: a model and an application. *Information Processing and Management* 41 (2005) 5-21, 2005.
15. Sowa, G.; Śnieżyński, T.; 2007. Technical Report (Deliverable) D61.4b - Security framework and architecture.
16. Tramontin Jr., R. J.; Rabelo, R. J. A Knowledge Search Framework for Collaborative Networks. In: *Proceedings of PRO-VE'2007 – 8th IFIP Working Conference on Virtual Enterprises*, 2007.
17. Uren, V.; Cimiano, P.; Iria, J.; Handschuh, S.; Vargas-Vera, M.; Motta, E.; Ciravegna, F. Semantic annotation for knowledge management: Requirements and a survey of the state of the art. In: *Journal of Web Semantics*. 4(1): 14-28, 2006.