Integration of an Ontology Manager to Organize the Sharing of Learning Objects in a Peer-to-Peer Network

Gautier Bastide, Pierre Pompidor, Danièle Hérin, Michel Sala

Integration of an ontology manager to organize the sharing of learning objects in a peer-to-peer network

Gautier Bastide, Pierre Pompidor, Danièle Hérin, Michel Sala
L.I.R.M.M.
Computer Science, Robotics and Microelectronics Laboratory
161 Rue Ada, 34392 Montpellier
{bastide,pompidor,dh,sala}@lirmm.fr

Abstract

This paper is dedicated to the ontologies management, and more particularly to a tool called Kaon. One of its objectives is to propose the integration of a such application in a peer-to-peer platform. Indeed, the tools which are provided by Kaon can be used for the management of the distributed resources sharing. The integration of Kaon to modelize and to organize the knowledge in the Edutella network, for example, would allow the realization of more effective search engines. The Kaon platform is already used for the annotations management in Edutella project. It would be judicious to spread it to all the resources which are shared within the Internet network. The use of Kaon would allow to increase the efficiency of searching services thanks to the modeling of the semantic links between the various learning objects which are shared through the Edutella network.

Keywords: peer-to-peer, Edutella, metedata, RDF, Kaon, ontology.

1. Introduction

The various search engines which are proposed and implemented within the framework of the Edutella project [Edutella; Nejdl et al., 2002] provide the user with a disorganized list of results. These work in the following way: the user enters a series of keywords or a request written in a language like Datalog or in RDF-QEL-i. Then the application passes on the requests, returns the results and present them to the user shape of a list of resources which it is possible to consult by select on. Every answer is described by metadata. The problem lies in the fact that the data shown in the screen are not organized.

Furthermore, the relations between the various solutions which are proposed to the user are not valorized within the framework of their visualization. This inconvenience meets itself in the major part of the traditional search engines. It would be more judicious to advance the existing links between the various resources found by the search engine so as to facilitate the reading and the interpretation of the results. This proposition requires the use of an ontology management tool such as Kaon [Kaon; Volz et al., 2003; Bozak et al.,2002] to modelize the links between the resources. A Kaon server must be present in each super-peer of the Edutella network.

A first proposition would consist in showing the results using an ontologies graph which is similar to that presented in one of the interfaces of the Kaon API i.e. OI-Modeler [OI-Modeler, 2002]. The graph presented to the user would be simplified but would allow him to navigate through the various concepts present in the answer. So the use of Kaon to model the links between the resources allows us to increase the efficiency of the search engine and facilitate the legibility of the answers.

As regards the interface, the user must have three possibilities of action at the level of the search for information in the Edutella network:

- Enter a series of keywords judiciously chosen,
- Load a request written in a language such as Datalog or RDF-QEL-i,
- Load an ontology which can be described in XML or in RDF.

Furthermore, the user must have the possibility of selecting a particular supplier of resources. If any Edutella supplier isn't selected, then the request will be spread to the whole network. The fact of giving the possibility to the user to select one or several Edutella suppliers of resources allows us to restrict the searches. Besides, if he wishes, the user has the possibility of querrying a particular supplier. He can also spread his searches to a series of suppliers that he has selected beforehand according to his preferences (if he wants to omit some suppliers).

2. Management of the metadata about the learning objects

The use of metadata is needed for the efficiency of the system of search for information. All the learning objects intended for the sharing must be described by metadata which respect predefined schemas such as those which are proposed by IEEE LOM, IMS or SCORM. In this part, we consider only objectives annotations. There are two possibilities to annotate a document (i.e. To field corresponding to the metadata allowing us to describe this resource). Either it is the creator of the educational resource which fills fields allowing us to describe it, or this stage is automatically made through tools of automatic anotations. The key points of this part are the maintenance of these metadata and the search for semantic links between the various resources.

The lack of annotations on a learning object could be resolved by analyzing the contents of documents to complete the missing data. This method of distribution of metadata by the analysis of the bibliographical references can be applied to one or several documents selected beforehand. It acts recursively. The point of departure for the application of this method is a document. Then, the method pursues its search in the resources quoted as references of the analyzed document. To avoid problems, it is thus necessary to introduce the notion of life cycle (Creation of an indication TTL). It allows one to stop the process beyond a certain number of documents which are gone through and so to avoid the infinite loops. Indeed, that this method works correctly, it is indispensable to have a core of resources which are manually annotated by their owner (Figure 1).

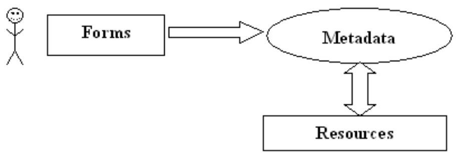

Figure 1. Creation of a core of annotated resources

At first, it is necessary to analyze the semantic contents of the document in order to establish a hierarchy of this component. In fact, the objective is to extract the ontology of the document. The metadata which are stored in the nodes allow to describe the various parts of the resource which is analyzed. A semantic link that must be created between two resources which are referenced in the same chapter of a document.

The completion of the annotations contents by this method of metadata propagation brings the creation of two main cases:

- If the resource is not annotated.

In that case, it is necessary to look for the references supplied by the authors. They are generally situated at the end by the document in the bibliography. Then, it is enough to retreive the corresponding informations. Thus, the annotations of a resource will be made thanks to the metadata describing the documents which are referenced. This operation necessarily arouses the creation of a semantic link between both resources. In that case, the reach of the method is global because the completion of metadata is made at the level of the document. A part of the information so harvested serve for describing the whole resource. It is necessary to retreive the part of the ontology of the resource 1 on which is going to be put the metadata which correspond to the ressource 2.

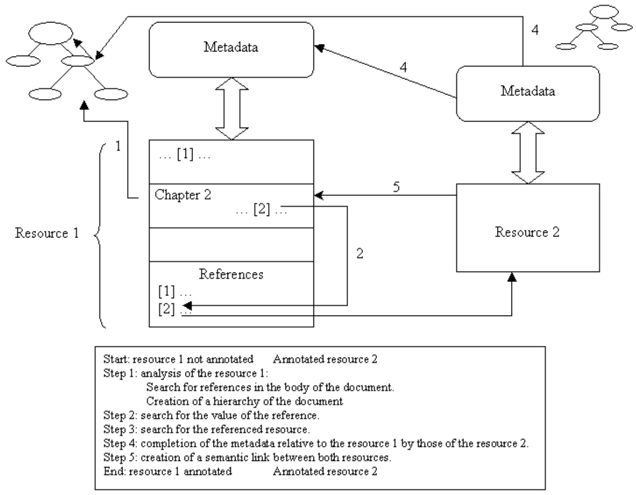

Figure 2. Annotation of learning objects: case n°1

- If the resource is already annotated.

In that case, it is enough to parse the document and to look for the references of the educational resources which are not annotated. When a resource is found, it is possible to fill in the information contained in its metadata with the corresponding data in the chapter being analyzed. A semantic link is then created between these two learning objects. In that case, the reach of the method is local because only a part of the learning object can be described by the metadata so collected. Only the information which correspond to the chapter of the annotated resource can be used to describe the new document.

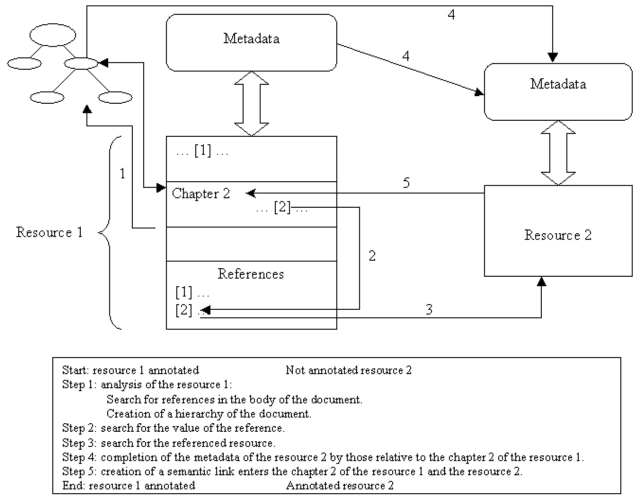

Figure 3. Annotation of learning objects: case n°2

All the metadata can't be propagated. Indeed, only those which correspond to the contents of the resource as the subject, the description, the keywords, the references can be duplicated. Other metadata (author, date of creation or publication, language) are not used within the framework of this method of propagation. The quantity of keywords obtained by this method can be quickly very important. Furthermore,

some of these words can have no link with the resource for which we try to annotate. The problems so met in this method of propagation of the metadata are owed to the evolution and to the revision of the ontologies. To manage all these problems, it is indispensable to set up certain number of operations intended to manipulate the ontologies relative to the description of the documents for which we try to annotate.
A first solution of this problem would consist in making the intersection of keywords relative to the resources pointed by this one. This method would allow to reduce the quantity of keywords being of use as description to the resource and would get a better result in their links with the document. This strategy must not be only applied to keywords but also to references. This method using the intersection of keywords does not allow to resolve all the problems relative to the fill of the metadata.
The methods of generalization on the local ontologies or external can also be a solution of the problems of metadata management.
Another solution of the evolution and the revision of ontologies would consist in using logic languages which allow to create inferences. The aim is to manage the data inconsistency.
Another problem which arises in the annotations management concerns the validity of the information contained in the metadata allowing us to describe the learning objects. Indeed, an annotation can be false (that is inappropriate in the resource or inaccurate). The problem can be due to incorrect data entry made by the resource's owner or to an error at the level of the automatic creation of annotations. It is possible to solve this problem by the use of a system of level-headednesses attributed to every resource's annotation. The users of the Edutella network have to have the possibility of acting on the value relative to the validity of the information contained in the metadata describing a resource. These operations can be made through a check by the users at every consultation of a resource. If an annotation is considered incorrect, either the owner of the resource is warned then he can operate modifications, or the mechanism of automatic creation of annotations computes again the information contained in the metadata of the resource.

3. Management of the ontologies at the level of super-node

The Kaon platform is used to manage the ontologies contained in the Edutella network. A Kaon server must be present on every super-node of the network. It is intended to manage the resources of its cluster and is necessary to know the big subjects of the other groups of peers.
The learning objects which are stored in the peers of the Edutella network are described by metadata and are grouped together in respect to ontologies. Indeed, they are considered as instances of concepts or sub-concepts. Every time a peer gives a resource, it must be declared to the Kaon server situated in the super-node to which it is connected. A new instance which square with the new ressource will be created. The Kaon API is then going to be in charge of integrating the resource into the existing model by taking into account semantic relations.
First of all, the Edutella supplier peer creates a description of its capacities as well as resources which it suggests sharing in the form of an ontology. Indeed, the resources which the peer suggests supplying must be described by metadata which respect standards of type IEEE LOM or SCORM and must be organized in the form of ontologies containing concepts, sub-concepts and instances. These correspond to the resources. Every instance possesses properties described by metadata and have relations with the other entities of the model. Every Kaon server has to manage at least two ontologies (figure 1): one allowing us to store the characteristics of the peers which are contained in the cluster as well as super-node of the Edutella network and another designed to store the metadata being of use as description to the learning objects and allowing the modeling of the semantic links between these. The information concerning the capacities of every Edutella supplier is stored in the Kaon server.

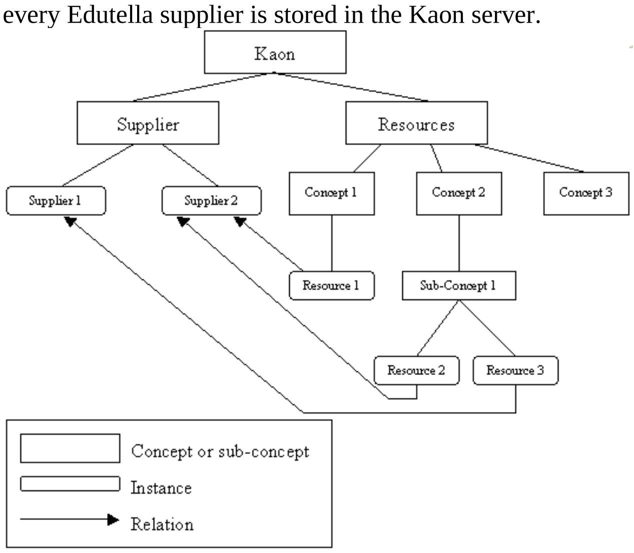

Figure 4. Example of an ontology model for the super-peers' management

The management of the knowledge is made on three levels which are interconnected:

- The first level: Learning objects.

It is the lowest level of the data model. It concerne the storage of the learning objects without metadata.

- The second level: metadata

This level contains the descriptions of the educational resources. The metadata generally follow a schema which is defined by standards such as IEEE LOM or SCORM.

- The third level: ontologies

This level contains the representation of the concepts, the sub-concepts and the links. This part allows one to organize and to manage components contained in the previous two levels. The instances of the ontology model contain the metadata (Level 2) which are used to describe the learning objects (Level 1). The learning objects (Level 1) are de-

scribed by metedata (Level 2) and regrouped by ontologies (Level 3).
The main relations which arise in ontologies of learning objects are the following ones:

- *Be_a_part_of*

The relation *Be_a_part_of(x,y,i)* means that x is a part of y. Thus, it is necessary to know the resource x if we want to study the resource y.
The value i represents the validity index of the relation (i.e. Reliable indication of the relation). In fact, it is a weight. This value has the same signification in the three following relations.

- *Be_explained_by*

The relation *Be_explained_by(x,y,i)* means that the resource x can be explained by the resource y.

- *Be_required*

The relation *Be_required(x,y,i)* means that the resource x needs the resource y as pre-required.

- Be_suggested

The relation *Be_suggested(x,y,i)* means that it is better to know the resource y before making the learning of the resource x. If you are interested in the resource x you can use it independently of the resource y. You don't have to know both resources.

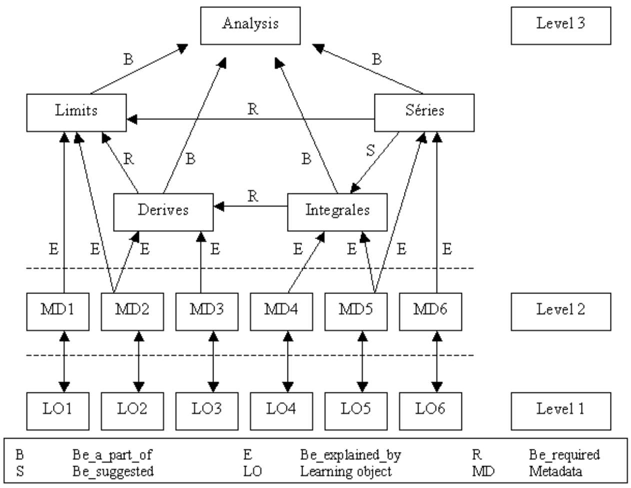

Figure 5. Example of an ontology model for mathematics [WebLearn, 2004]

The references supplied by the authors must be used to create semantic links between two resources. If a link doesn't exist between these two resources, a relation of type " Be_suggested" will be created. In this case, it is indispensable to create a relation which is the most flexible possible when we don't know the exact kind of link between two resources. Moreover, the kind of relation must be able to be modifiable by the authors of the resources. The goal of this operation is to improve the semantics of the model.
One of the most important points in the management of ontologies lies in their maintenance and in their evolution. Indeed, the model must be able to evolve every time that a new supplier connects to the network or every time that a peer share a new resource. The ontologies have to remain viable at all times. For that purpose, the Kaon API possesses numerous features which allow us to resolve this kind of problem. The updates must be made automatically while verifying the integrity of the model. It is indispensable to be able to introduce new concepts or even new relations into the existing model. For that purpose, it is necessary to be able to discover that the new resources really introduce new concepts and to determine their positions within the model. These operations must be automatically realized through syntactical analysers.

4. Scenario of connection of a peer supplier

The peer which tries to join the Edutella network looks for the super-node which appears to be the most suited to the fact that it suggests supplying as learning objects (Figure 6). For that purpose, the supplier peer is going to send a message to any super-nodes (which is used as a point of entry to the Edutella network). It will transmit the message which contain the informations about the new learning objects which are proposed to other super-peer of the Edutella network. When the super-node comes up to the expectations of the supplier peer, this one is going to be bound there. Then it will be sent to it all the information which it has collected beforehand. Once these operation made, the Kaon server which is situated on the super-node selected is going to have to update its model and so integrate the ontology proposed by the new peer. The lexical data must be stored in the Kaon server. The fusion of these ontologies is realized by the Kaon API and is made thanks to the tools which it possesses.

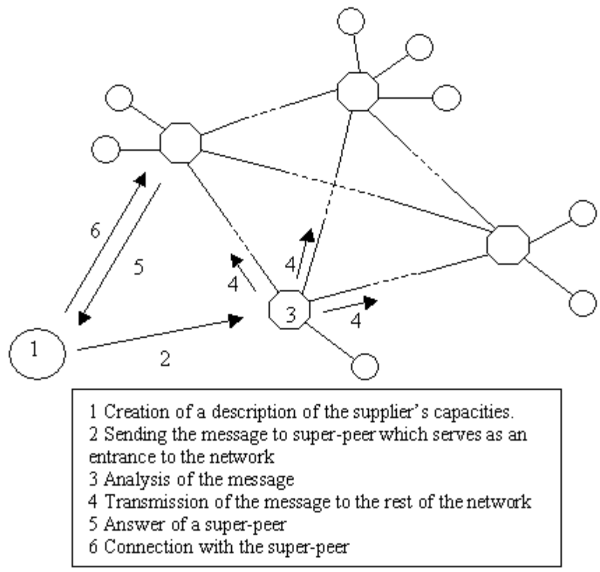

Figure 6. Scenario of connection of a new Edutella supplier peer

5. Scenario of search for information in the network

Here is an example of scenario of search for information in a Edutella network using the Kaon platform to describe the learning object and modelize the links between the entities of the model.

The user enters a series of keywords, either loads a request written in languages such as Datalog or RDF-QEL-i, or even an ontology which can be described in XML or RDF. If the user loads an ontology, the search engine takes care to complete it by adding instances, sub-concepts as well as relations between the various entities of the model.

The peer translates the request into the basic language of a Edutella network i.e. RDF-QEL-i. Then it passes on it in its super-node. This is then going to query the Kaon server through this language of request. Thanks to the data stored by the Kaon server, it is possible to determine which peer of the cluster or which other super-node contain the informations which we are looking for. Thanks to the data so harvested and to the dynamics tables of routing, the request can be passed on to the peers who may supply a correct answer.

If no good result is obtained, the request is passed on in the other nodes of the network.

Otherwise, it is going to send back the result of the request to the peer of origin in the form of a graph of RDF data.

In this way, when a super-peer receives the request, it requests to the Kaon server which manages its resources and to send back if possible a RDF graph in reply.

When the peer receives the answers, it collects and reorganizes the graphs obtained so as to be able to present to the user an unique graph allowing one to show the links between the resources and the various concepts included in the result of his request.

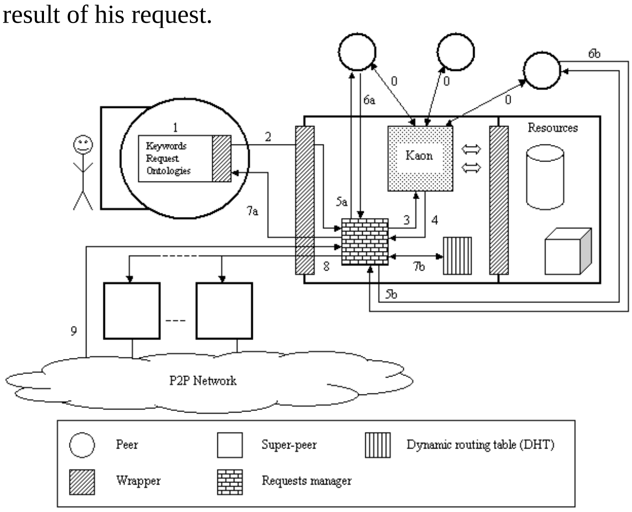

Figure 7. Scenario of Kaon integration in an Edutella network

The senario follow this model (Figure 4):

Step 0: stage of initialization (cf. 4. Scenario of connection of a new supplier peer).

The peers have to communicate with the Kaon server of the super-node to be able to modelize the semantics links between the resources which are shared at the level of the cluster. The resources must be described as instances of concepts or sub-concepts.

Step 1:

The user creates or loads a request which is going to be spread in the Edutella network. Several methods are possible: either the user supplies a list of keywords judiciously chosen, or he directly loads a request which he has beforehand written in a language such as Datalog either RDF-QEL-i with i lower than 5, or he defines an ontology (Composed of concepts, by sub-concepts, by properties and by instances) written in a language as RDFS or OWL. Another possibility would be to propose to the user certain number of subjects and sub-subjects which would allow him to obtain more general information. Furthermore, the user has to have the possibility of selecting the language in which he wishes to obtain the answers (The lexical information must be stored in the user's machine).

Step 2:

The request so formed is translated into a language (RDF-QEL-i) which is understandable by all the components of the Edutella network (Peers and super-nodes). The data are passed on in the super-node which is under the responsibility of the peer of the original request. This is going to be retreived by the administrator of request of the super-node which then undertake to do its treatment.

Step 3:

The request retreived by the administrator is then going to be modified to be able to request to the Kaon server containing all the information necessary for the management of the ontologies modeling the cluster. Kaon can resolve requests in the RDF-QEL-i language.

Step 4:

The Kaon server returns the result of the request to the super-node's administrator to know the list of peers of the cluster which may resolve the request. Two cases are then possible:

- If no result is found by the server, then the administrator of requests consults the dynamic tables of routing (***Step 7b***) to obtain the address of the super-node being able to solve the request. This request is going to be passed on in the super-peer so found (***Step 8***).
- Otherwise, the request is passed on to the peers which are so found (***Step 5a***, ***Step 5b***). The peer are going to resolve locally the request and to send back the result under the shape of a RDF graph to the super-node (***Step 6a***, ***Step 6b***). The data thus retrieved is analysed by the administrator. Two cases appear:
  - Either, the results obtained are satisfactory (According to a certain number of pre-defined criteria),
    In that case, the data which are collected by the administrator is sent to the original peer (***Step 7a***).

- or the results do not allow us to propose a correct answer.

In that case, the request is passed on to the other super-node of the Edutella network thanks to the dynamic tables of routing (***Step 8***).

Step 9:

The results of the request are then passed on to the broadcasting peer of the request to be collected. The transmission of data answers is made through super-peers of the Edutella network. The results can be presented to the user. They are organized by ontology. The semantics links between the resources are shown in such a way that the user can find quickly what he is looking for.

The integration of the Kaon plaform inside the Edutella super-peer allow us to improve the searching service. This tool is presented as a complement to the dynamic routing tables. The learning objects are described by metadata and are grouped by ontologies. The goal is to retrieve it and locate it easily and efficiently.

6. Conclusion

This article proposes the use of an ontology administrator like Kaon to modelize the links between resources which are shared within a peer-to-peer network like Edutella. The use of such a tool allows to improve the service of search for information of the Edutella peer-to-peer network. Indeed, the knowledge concerning the resources contained in the super-nodes of the network are grouped together by ontologies so as to be able to localize them more easily and more quickly. The scenario proposed in this paper describes the various phases relative to the implementation of an ontology management tool within super-peers. The integration of this type of tool to manage the resources is an undeniable contribution in the field of the information search within a totally decentralized environment. It allows one to model better knowledge contained in a network as well as their relations. This tool is presented as a complement to the dynamic tables of routing in the optics to more easily and quickly track down and localize one or several points of information which the user is looking for.

References

[Edutella] Project Edutella. http://www.edutella.jxta.org/

[Nejdl et al., 2002a] W. Nejdl, B. Wolf, C Qu, S. Decker, M Sintek, A Naeve, M. Nilsson, M. Palmér, T. Risch. EDUTELLA: A P2P networking infrastructure based on RDF. 11th International World Wide Web Conference, Hawaii, USA, May, 2002.

[Nejdl et al., 2002b] W. Neijdl, B. Wolf, S. Staab, J. Tane. EDUTELLA: Searching an Annotating Ressources within an RDF-based P2P Network. Semantic Web Workshop 2002 Honolulu, Hawaii, May 7, 2002.

[Nejdl et al., 2003] W. Nejdl, W.Siberski, M. Sintek. Desing Issues and Challenges for RDF- and Shema-Based Peer-to-Peer System. ACM SIGMOD Record, Volume 32, Issue 3, September 2003.

[Kaon] Project Kaon. http://www.kaon.semanticweb.org/

[Volz et al., 2003] R. Volz, S. Staab, D. Oberle, B. Motik. KAON SERVER - A Semantic Web Management System. Twielth Internationnal World Wide Web conferance, WWW 2003, Budapest, Hungary, May 20-24, 2003. ACM, 2003.

[Bozsak et al., 2002] E. Bozsak, M. Ehrig, S. Handschuh, A. Hotho, A. Maedche, B. Motik, D. Oberle, C. Schmitz, S. Staab, L. Stojanovic, N. Stojanovic, R. Studer, G. Stumme, Y. Sure, J. Tane, R. Volz, V. Zacharias. KAON – Towards a large scale Semantic Web. E-Commerce and Web technologies, Third Internationnal Conference, EC-Web 2002, Aix-en-Provence, France, September 2-6, 2002, Proceeding, volume 2455 of Lecture Notes in Computer Science, pages 304-313, Springer, 2002.

[Kaon, 2004] KAON: The Karlsruhe Ontology and Semantic Web Framework. Developer's Guide for KAON 1.2.7, January 2004. http://www.aifb.uni-karlsruhe.de/WBS

[OI-Modeler, 2002] OI-Modeler User's Guide. November 27th, 2002. http://www.aifb.uni-karlsruhe.de/WBS

[WebLearn, 2004] Workshop organized by the action « Semantic Web and E-Learning » of Kaleidoscope NoE, Paris, May 3-4, 2004.