Abel, M.-H., Benayache, A., Lenne, D., Moulin, C., Barry, C., & Chaput, B. (2004). Ontology-based Organizational Memory for e-learning. *Educational Technology & Society*, 7 (4), 98-111.

Ontology-based Organizational Memory for e-learning

Marie-Hélène Abel, Ahcène Benayache, Dominique Lenne, Claude Moulin
UMR CNRS 6599 Heudiasyc, BP 20529
University of Compiègne
60205 Compiègne Cedex, France
Tel: +33 34423 4950
Fax: +33 34423 4477
Marie-Helene.Abel@hds.utc.fr
Ahcene.Benayache@hds.utc.fr
Dominique.Lenne@hds.utc.fr
Claude.Moulin@hds.utc.fr

Catherine Barry
FRE CNRS 2733 LARIA
Université de Picardie Jules-Verne
80000 Amiens cedex, France
Tel: +33 32282 5410
Fax: +33 32282 5412
Barry@laria.u-picardie.fr

Brigitte Chaput
Equipe SaSo, groupe NTE
Université de Picardie Jules-Verne
80000 Amiens cedex, France
Tel: +33 2272 6980
Brigitte.Chaput@u-picardie.fr

Abstract

e-learning leads to evolutions in the way of designing a course. Diffused through the web, the course content cannot be the direct transcription of a face to face course content. A course can be seen as an organization in which different actors are involved. These actors produce documents, information and knowledge that they often share. We present in this paper an ontology-based document-driven memory which is particularly adapted to an e-learning situation. The utility of a shared memory is reinforced in this kind of situation, because the interactions do not usually occur in the same place and in the same time. First we precise our conception of e-learning and we analyze actors needs. Then we present the main features of our learning organizational memory and we focus on the ontologies on which it is based. We consider two kinds of ontologies: the first one is generic and concerns the domain of training; the second one is related to the application domain and is specific to a particular training program. We present our approach for building these ontologies and we show how they can be merged. Finally we describe the learning memory and the prototype we realized for two course units proposed in our universities.

Keywords

E-learning, Ontology, Organizational memory, Topic maps.

Introduction

Information Technology has already transformed the way people work and has an increasing impact on the long life learning. New approaches mainly based on the utilization of web technologies are proposed. They often refer to the concept of "e-learning". Unfortunately, the term e-learning is used to designate various types of situations such as administrative course management, web-based learning, or videoconferences. Usual e-learning definitions put the emphasis more on network utilization and pedagogical content than on distribution of courses. Numerous documents resources may be used during e-learning. Some are internal and made by several actors implied in the e-learning. Others are available on the web: on-line courses, course supports, slides, bibliographies, frequently asked questions, lecture notes, etc. The increasing number of available resources is a real problem in content management systems.

Research work on the Semantic Web aims at addressing this kind of problem. The Semantic Web is an extension of the current web in which information is given well-defined meaning, enabling computers and people to better work in co-operation (Berners-Lee et al., 2001). The idea is to represent web data, to define and link them so that they can be used for more effective discovery, automation, integration, and reuse across various applications. The Web can reach its full potential if it becomes a place where data can be shared and processed by automated tools as well as people. Sharing data is one of the basic principles the Semantic Web will operate on. In order to be able to exchange the semantics of information, one first needs to agree on how to explicitly model it. Ontologies are a way of representing such formal and shared information. They can be used to index data indicating their meaning, thereby making their semantics explicit and machine-accessible.

In the MEMORAe project, which relates to this research field, we propose to consider an e-learning training as an organization and to manage the resources of this organization by the means of an ontology-based "learning organizational memory" (Abel, Lenne, & Cissé, 2002). This memory allows, on one hand, to capitalize the learning knowledge, and on the other hand, to better index resources, taking into account the learning context. The MEMORAe project (that stands in French for Organizational Memory Dedicated to e-learning) is supported by STEF, a research pole of the French Picardie region.

In this paper, we first present our conception of e-learning thanks to a scenario of use. Then we introduce the notion of "learning organizational memory". This notion requires specific ontologies that allow to organize knowledge element and index resources. We introduce two application ontologies about algorithms and statistics. They are originally based on the experience of two courses: "Algorithms and Programming Pascal" (NF01) and "Statistics" (B31.1), respectively taught at the French universities of Compiègne and Amiens. Building such ontologies is a hard work and we propose to illustrate the method we followed to build them. Then, we show that it is necessary to merge an application ontology into a domain ontology and the way it is concretely possible. Finally, we illustrate the use of the memory through a prototype we have developed and still maintain.

e-learning

Our conception of e-learning

The term 'e-learning' is currently very used and refers to various notions such as logistic (administrative management), resources (course broadcasting) or technology (virtual conference tools). Numerous definitions of e-learning have been proposed. They usually put the emphasis on network utilization (explaining the « e » in e-learning) and on Information Technology. E-learning must not be reduced to the use of new technologies to serve old learning modes. It is supposed to lead to new learning forms. This implies some consequences. For example, e-learning needs at least:

- A reflection on the content: goals, concepts to study, competences to acquire, etc.
- A reflection on the content organization: relations between learning concepts,
- A construction of new resources taking into account possibilities offered by Information Technology: direct digitalization of old resources is not sufficient,
- A redefinition of actors (teachers, learners) roles.

Within the MEMORAe project, we are interested in the building of a pedagogical content under a granular form represented by ontology of concepts. Users must have free access to this ontology. Indeed, we consider that the learner must have an active role in his learning. Available documents are not simply transcription of classical courses. They consist in a set of resources that intend to be easy-to-access because of their indexation by the ontology of learning domain concepts. The courses we deal with are scientific courses taught at university.

Use scenario

In our conception of e-learning, knowledge and information structuring is central as well for learners as for teachers. The ontology-based organizational memory we propose aims at helping them to structure and manage knowledge related to a given course or training unit. It relies on an organization model of this course unit and takes into account teachers and learners viewpoints.

In an e-learning situation, learners are often geographically distant. It is thus necessary for them to have an easy access to documents and more generally to resources they need. But because of the distance, they often need to get into contact and to dialogue with teachers and with other learners. Furthermore, certain types of activities (such as practical work) explicitly require cooperation between students.

During training, learners are often led to ask questions regarding the content of a course. For example: What are the goals of this lesson? What are the notions to be learnt? What are the prerequisites? Is there any order in these notions? Are there any documents to consult (slides, books, etc.)? What is it possible to do in order to improve a lesson? Is there any web site, newsgroup dealing with this lesson? Etc.

During training, students have often to produce documents that are sent to teachers for evaluation or that are kept. In this last case, documents can be for example work or synthesis documents or annotations (Marshall, 1998). The students can decide (or propose) later to make these documents available for other users. It is therefore useful to allow the attribution of different grants to documents.

The definition of a shared vocabulary is a key point in order to facilitate access to documents, dialogue with teachers and collaboration with other learners.

Learning organizational memory

A course unit is based on knowledge and competencies it should provide, on actors (learners, instructors, trainers, course designers, administrators, etc.) and on resources of different types (definitions, exercises with or without solution, case studies, etc.), and different forms (reports, books, web sites, etc.). In this sense, a course is an organization.

A common approach to tackle the knowledge management problem in an organization consists in designing an organizational memory. Such a memory can be seen as “an explicit and persistent representation of knowledge and information in an organization, in order to facilitate their access and reuse by members of the organization for their tasks” (Rabarijaona et al., 2000).

An organizational memory allows capitalizing not only pedagogical resources related to the contents of the course but also information on actors themselves (specificities, background, profile, etc.). It allows administrative management (registration, notes, etc.) of the course too.

In order to share information in an organization, actors have to use a common terminology, especially when they are geographically distant. A given word or expression must have the same meaning for everyone. It is one of the reasons why organizational memories are often based on ontologies.

Organizational Memories and Learning Organizational Memories

A learning organizational memory is different from an organizational memory because of its goal, which is to provide users with content and more precisely pedagogical content. This pedagogical content is composed of the notions to acquire, the links between these notions and the resources they index.

Notions are not only chosen because they are related to the course unit, they are also the result of a reflection on the course itself. A pedagogical work has to be done. For example, with NF01, why and how to make a link between the “loop” and “array” notions?

Resources have to be selected relying on pedagogical goals. The choice of their indexation terms is related to this goal too. It is not an automatic indexation. The course manager (with the help of an editorial committee if needed) is responsible for the pertinence of the links. It is not because a document treats of a notion to acquire that it will be necessary indexed by this notion. The choice is explicit, that is to say that the document must have been evaluated as sufficiently adapted to the learning of this notion.

These choices are part of the pedagogical scenario the course manager wants to implement. In a classical organizational memory, there is no pedagogical scenario because the objective of this kind of memory is not training.

The learning organizational memory we propose aims at facilitating knowledge organization and management for a given course or training, and at clarifying competencies it allows to acquire.

Pedagogical content of a learning OM

The pedagogical content of a learning organizational memory is mainly composed of the notions to learn, the links relating them and the documents indexed by them. The manager of a training memory is responsible for its content, that is to say the choice of the notions to learn and the documents indexed by these notions. In this sense there is no course design (as it can exist in a linear course), but more precisely pedagogical content selection. We precise below what are "notions to learn" for us, then we present different kind of pedagogical resources and our conception of annotations.

Notion to learn

The design of an e-learning application implies to focus on the learner, giving him/her the means to be active, to make him/her understand the resources that are at his/her disposal and to teach him/her how to search and to use them. Articulating a course starting from knowledge grains offers more individualization possibilities. For some authors (Boullier, 2001), it consists in dividing the course content in fine grains, using a semantic mark-up.

On the contrary, we do not use the expression 'notion to learn' to refer to a course unit part, but to a notion to acquire. Consequently, there is no need to cut off existing documents or to produce new documents corresponding to these notions. Authors remain free regarding the making of their documents. They do not have to follow graphical or contents guidelines. Moreover they can reuse existing documents.

Notions to learn (i.e. fine grains) are used as indexes to access documents related to them. A notion to learn can refer to several documents (giving several means to acquire it) and a document can be referred to by several notions (giving several means to retrieve it).

Note that a digital document can be already made of several parts that can be themselves indexed. It will however remain a document as a whole for which the author has no writing guidelines to follow. Furthermore another logical partition of this document can be done by the author or the editorial committee later.

Pedagogical resources

Pedagogical resources are generally documents: course texts, course notes, slides, e-books, reports, books presentations, links to web sites … Among the represented documents, some (digital documents) are stored in the memory and others are references to physical documents.

Resources can be accessed according to different rights. They can be private. In this case, users only store them in the memory and do not want to give other users access to them. They can be annotations, work in progress, downloaded and not yet analyzed documents … Resources can also be semi-public or public, that is to say shared by part or all of the users. For example, an annotation of a reader giving his/her motivated impression on a document can help memory users to choose appropriate documents. Moreover, several annotations written by different authors or relying on different notions can be attached to a same document.

Resources can also have different status. They can be terminated and validated documents, or on the contrary, working documents written by one or more users and therefore shared by them during the time of their realization.

Annotations

Our reflection on annotations started from two observations:

- On one hand, when users of the memory access a notion to acquire, there are faced with several resources related to this notion. The choice can be based, as it is presently, on several associated characteristics: author, resource type (book, web site, etc) but it could be guided by other information such as comments or remarks on the resources.
- On the other hand, the role of an organizational memory is to capitalize knowledge. It is then useful to keep track of the reasons that led a course manager to choose a resource, a notion, or a link between two notions.

We propose to take into account this information by using annotations. (Marshall, 98) identified different dimensions allowing to characterize different approaches of annotations:

- Formal vs. informal annotations. Examples of formal annotations are metadata, specifically metadata that follows structural standards and are assigned values using conventional authorities.
- Explicit vs. tacit annotations. Many personal annotations, by their nature, are telegraphic, incomplete, and tacit. On the other hand, annotations written for others are usually more explicit.
- ·Permanent vs. transient annotations. Annotations may not be permanent. If annotations are reflections of a reader's engagement in the text, their value may only hold for the current traversal through the narrative or hyper narrative. On the other hand, some annotations have been observed to bring value to future readers.
- Published vs. private. We all know of circumstances in which annotations are private form. On the other hand, annotated editions of important scholarly works are a good example of published commentary.

In the Memorae project we consider that an annotation:

- Is a resource, result of an annotation action.
- Is related to a target that can be a notion to learn (concept), a link between concepts, a resource, a part of resource, a collection of resources.
- Has one or several authors and presents its/their comments on the target. These comments are created at a given date, with a precise objective, and are directed to a precise audience (that can be the author himself in case of a personal annotation).
- Is not part of the target itself. It is then necessary to make a link between the target and the annotation.
- Makes sense only in its context (target, author, goal, audience).
- Can be text, graphic, voice or illustration.

Note that a target must have a representation in the memory, in order to be annotated. As an annotation is a resource, it can be itself annotated. Following this conception our notions to learn are not annotations, they are metadata. We will now see how we represent them using ontologies.

Ontologies

Ontologies for e-learning

For navigating through the memory, the end-users (learners, teachers, etc.) need a shared vocabulary. That is why we decided to model the memory with ontologies. From the different ontology types defined by Van Heijst (Van Heijst, 1997), generic ontologies, domain ontologies, application ontologies and meta-ontologies, we only use the second and third categories. We have to consider two aspects for modeling the memory and building ontologies (Breuker et al., 1999). First the domain of training has its own characteristics. Secondly, it must be linked to the application domain of a particular training program. According to Gruber (Gruber, 1995) "An ontology is an explicit specification of a conceptualization". Guarino gives a precision on this definition, considering that ontologies are necessarily a partial specification of a conceptualization (Guarino, 1995). We can add with Gruber "an ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents".

The first ontology (domain ontology) we have to specify, describes the concepts of the « training » domain. They can be users types (tutor, secretary), documents types (book, slides for oral presentation, web page, site, etc.), media types (text, image, audio, video). They can also be pedagogical characteristics (activity type) and they can refer to point of view (annotation). It is difficult to directly reuse part or a whole of existing ontologies because they mainly depend on objectives and choices for specific needs, but we must consider the help they can bring.

We studied pedagogical ontologies like the one presented in (Chabert-Ranwez, 2000). It deeply describes types of tests as multiple choice, true false, or matching. It is also possible to find activity types description in (Desmoulins & Mille, 2002). Pedagogical resources are not organized following the way recommended by the Learning Object Metadata standard in the Educational Category, because we do not agree to associate various activity types like exercise or simulation, with data representation like diagram, figure or graph in the same set. A description of the LOM standard can be found in the document 1484.12.1, http://ltsc.ieee.org/wg12/index.html.

The second ontology (application ontology) specifies the organization of theoretical notions that are studied during training session. In the example of an initiation to algorithmic, some notions like data structure or control structure are explained. It is possible, but not mandatory, to consider "tree" and "array" as sub-concepts of the concept “data structure” and to define the relation “uses” between the concepts “data structure” and “iterative structures” (in this case they are the domain and range value of this relation).

These ontologies are not independent; the second one is necessarily attached to the first one. For example, to express that a document is an introduction to data structures we join the two concepts “introduction” and “data structures” that do not belong to the same ontology. Pedagogical relations like “prerequisite” or “uses” that occur between concepts of the application ontology are defined in the domain ontology. However, specific roles can belong to the application ontology (for example for the B31.1 application, “has a cardinality”).

Integration of ontologies

In the MEMORAe project the domain is the training itself. Its corresponding ontology has to be linked to application ontologies. Figure 1 shows this integration. The root of the project ontology is memoraObject. First, this concept must be the root of all the concepts belonging to application ontologies. The sub-concept knowledgeBeanObject allows the integration of application ontologies. Their root concept must extend it.

Secondly, the memoraObject concept must also be the root of all the concepts that belongs to the training ontology. Its root is called here trainingOntologyObject. The project defines a special concept called knowledgeBean whose elements are the concepts of application ontologies. They are the notions that learners have to study in the training. This concept extends the specific trainingClass containing all the concepts of the domain ontology.

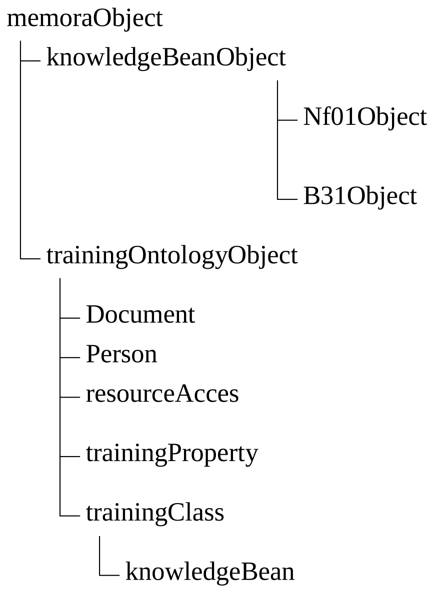

*Figure 1.*Integration of ontologies

Elements of the domain ontology

Figure 1 shows the upper elements of the domain ontology. We give in this section more details about it (see Figure 2). Actors of the training program are instances of the concept Person and we consider four categories listed in the figure. A person can also play a role in a relation: author, responsible, or tutor for example. Documents are organized according to their form, more or less structured. We present in Figure 2 the main categories. Each document is associated with a support (ResourceAccess in Figure 2), digital or not. Some elements listed above can be found in the LOM description.

- ResourceAccess
  - Digital
    - Audio
    - Video
    - Text
    - Image
  - Solid
    - Image
    - CD
    - Paper
    - Slide
    - Audiotape
    - Videotape
- Document
  - Picture
    - Graph
    - Diagram
  - Exercise
  - Message
  - Annotation
  - Lecture
  - Index
  - Slide show
  - Book
  - Web page
  - Web site
- Person
  - Administrative
  - Student
  - Technicien
  - Teacher

Figure 2. Elements of domain ontology

TrainingProperty in Figure 1 is the class of relations occurring between concepts. Some are more pedagogical as prerequisite for example. Other are more general as writtenBy that allows to link a document and a person. Some link names are coming from the Dublin Core initiative (http://dublincore.org/documents/dces/) about document indexation. Binary relations have a domain and a range for constraining instances of the relation, but more generally we can include a relation inside a Cartesian product of generic concepts. We write for example writtenBy Ì Document:work ´ Person:author to signify that the relation writtenBy occurs between individual concepts of the concept Document, playing the role of work and individual concepts of the concept Person playing the role of author. We write in the same way: prerequisite Ì knowledgeBean:learnBefore ´ knowledgeBean:learnAfter

When writing application ontology compliant with that domain ontology only few constraints appear:

- The root of the ontology must extend the concept knowledgeBeanObject as nf01Object and b31.1Object in Figure 1.
- Each concept of the application ontology must be an instance of the concept knowledgeBean.
- It is possible to use relations defined in the domain ontology.
- It is possible to create relation between concepts of both ontologies.

Populating the memory

We give an example of annotation to show the way the memory can be populated. An annotation allows to give a suggestion about either one concept or a set of concepts. In the last case, there is no particular relation occurring in the ontology between the concepts that must be annotated. It is not an annotation of each single concept but of the reunion of all. When navigating, it is important and necessary to get the annotation document from any concept that is concerned by this annotation. The domain ontology contains the suggestion_annotation relation defined by: suggestion_annotation Ì Bag:about ´ Annotation:information. Bag is a domain concept that allows to group knowledgeBean elements. Note that we also use this facility for annotating one concept. For example, if we want to give information about the use of the concepts set and complement defined in the statistics ontology, in the population we would have:

a. bag_1, instance of Bag
b. element(Set,bag_1)
c. element(Complement,bag_1)
d. ann_1, instance of Annotation (Annotation is a subclass of Document)
e. suggestion_annotation(ann_1,bag_1)

For indicating the author of the annotation:

f. Abel, instance of Person
g. writtenBy(ann_1,Abel)

A method for building ontologies

The analysis of several research works (Aussenac-Gilles, 2000; Bachimont, 2000; Gomez-Pérez, 1999; Kassel, 2002; Uschold et al., 1996) allows reaching a consensus on ontology building process. It relies on two steps: ontologization and operationalization (see Figure 3).

The ontologization step consists in building a conceptual ontology. Knowledge of a domain is elaborated in two ways:

- Human followed by machine analysis of various kinds of resources such as glossaries, books, courses, other ontologies, texts, etc., revealing terms and semantics structures (Fernandez et al., 1997).
- Expert interviews.

The operationalization step consists in coding the conceptual ontology using an operational knowledge representation language (i.e. equipped with inference mechanisms). This step can lead to loss of information.

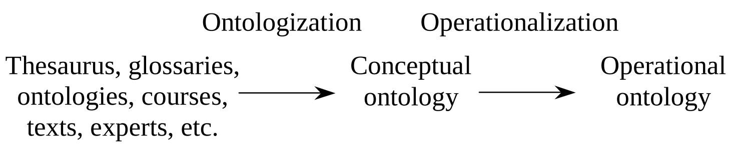

Figure 3. Two main steps in ontology building process

Concepts are often structured using taxonomies. In order to reveal various points of view in taxonomy, the OntoSpec method (Kassel, 2002) introduces the notion of semantic axes (SA). A semantic axe groups the sub-concepts according to their differentia (characteristics which allow to distinguish between sub-concepts). For example, in Figure 4, the set concept is specialized in three axes. The differentia of axe 1 is finite/infinite. In the same way, Guarino and Welty (Guarino, 1992; Guarino & Welty, 2000) propose criteria for properties classification in order to guide construction of taxonomy and to check subsumption links validity.

To build taxonomy of concepts, three approaches can be considered (Fridman & Hafner, 1997):

- Top-Down approach: first top-level ontology concepts are built, and then they are specialized.
- Bottom-Up approach: first low-level ontology concepts are built then they are generalized.
- Middle-out approach: first most important concepts are built, then they are generalized and specialized.

Within the MEMORAe project, we are interested in building a conceptual ontology. For the B31.1 application, we built the conceptual ontology in collaboration with an expert. In order to facilitate the access to the memory resources, we have to represent the notions to acquire according to the different points of view. For these reasons we chose to follow the OntoSpec specification method developed by the LaRIA IC team (http://www.laria.u-picardie.fr/equipe_ingenierie_connaissances.html).

OntoSpec specification method

OntoSpec is a method of semi-informal specification of ontologies (Kassel, 2002). It supposes that a conceptualization is made up of a set of concepts (or conceptual entities) and relations. The concepts in OntoSpec are organized in a taxonomy. Sub-concepts inherit all the properties of their super-concept. The relations make it possible to connect various concepts between them.

A conceptual entity owns a definition and denotes a set of objects having properties. The entity definition structure is based on a classification of these properties. At a first level, the properties are either Essential Properties (EP) or Incidental Properties (IP). The EPs are verified by all the objects denoted by the entity in every situation, or possible world. They are thus really definitional. Conversely, the IPs are satisfied only in a sub-range of situations. At a second level, the properties are classified according to roles they play regarding the conceptual entity. These roles can be abstract, e.g. Necessary Condition (NC), Sufficient Condition (SN), Necessary and Sufficient Condition (NSC). If the entity is defined by NSC, then its definition is complete. It is enough to characterize the entity.

An ontology is a differential set of concepts: the concepts are positioned according to their differences (Bachimont, 2000). In fact, the set of concepts are structured hierarchically and the properties are bound by conceptual properties. The conceptual property that structures a hierarchy of concepts is the subsumption, which binds two concepts: the concept C1 subsumes another concept C2, (respectively the relation R1 subsumes another relation R2, if and only if all instances of C2 are necessarily instance of C1. The sub-concept is more specific than the super-concept and denotes less amount of objects (smaller extension).

Sibling concepts are organized in semantic axes according to their similarities. The set concept is specialized according to three axes: finite/infinite, countable/uncountable, subset/superset (see Figure 4).

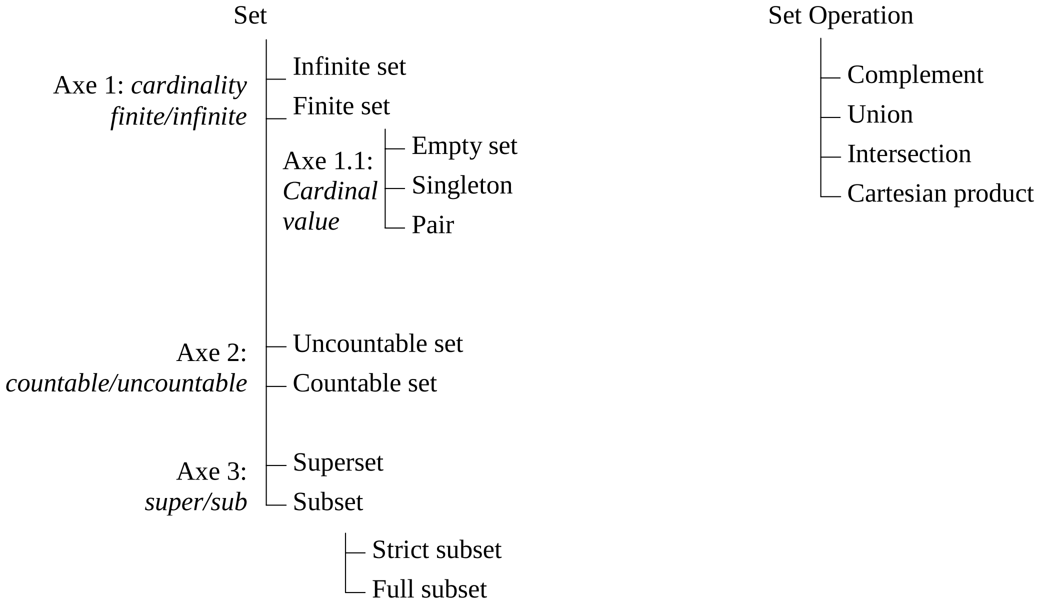

Figure 4. Specializations of the “set” notion

OntoSpec specification method also defines a list of the more specific properties that can be associated to a concept or a relation. This will not be developed here. It is semi-formal because it requires a definition of the conceptual entities (concepts and relations) using a strongly structured language. Figure 5 presents examples of definitions. The labels in square brackets correspond to the distinctions between properties; the underlined texts represent the relations.

Concepts:

- *Finite set*: [EP/NSC] A FINITE SET is a SET which has a cardinal. [EP/NC] Any FINITE SET is COUNTABLE. [PE/NC]. The FINITE SETS are DISTINCT from the INFINITE SETS.
- *Infinite set*: [EP/NSC] An INFINITE SET is a SET which DOES NOT HAVE a CARDINAL. [EP/NC] The INFINITE SETS are DISTINCT from the FINITE SETS.
- *Cardinal*: [PE/NSC] A CARDINAL is a NUMBER which COUNTS THE NUMBER OF ELEMENTS of a SET.

Relations: A CARDINAL counts the number of elements of a UNIT.

- *Has a cardinal* : [EP/NSC] To have a cardinal implies TO BE COUNTABLE. [EP/NC] A CARDINAL counts the number of elements of a SET.

Figure 5. Semi-informal definition of three concepts and of one relation

Application ontologies

We present in this section some excerpts of B31.1 application ontology starting from the definition of two concepts "set" and "operation":
Definition of the conceptual entities associated to the generic concept "set" (see Figure 4):

Concepts:

- SET ; collection of elements [SA]: The concept of SET specializes in FINITE SET and INFINITE SET, according to the relation: HAS A CARDINAL. [SA] The concept of SET specializes in COUNTABLE SET and UNCOUNTABLE SET, according to the relation: COUNTABILITY [SA]. The concept of SET specializes in SUBSET and SUPERSET, according to the relation: PART WHICH THE SET PLAYS COMPARED TO ANOTHER SET.
- Infinite set: [EP/NSC] An INFINITE SET **is a** SET which DOES NOT HAVE a CARDINAL. [EP/NC] The INFINITE SETS are DISTINCT from the FINITE SET.
- Finite set: [EP/NSC] A FINITE SET **is a** SET which **has a cardinal**. [EP/NC] Any FINITE SET is COUNTABLE. [PE/NC] The FINITE SETS are DISTINCT from the INFINITE SETS. [SA] The concept of finite set specializes in PAIR, EMPTY SET and SINGLETON, according to the relation: VALUE OF THE CARDINAL.
- Pair: [EP/NSC] A PAIR **is a** Set SET **has a cardinal** = 2.
- Empty set: [EP/NSC] An EMPTY SET **is a** SET which **has a cardinal** = 0.
- Singleton: [EP/NSC] A SINGLETON is a SET which **has a cardinal** = 1.
- Countable set: [EP/NSC] A COUNTABLE SET **is a** SET which elements can be numbered. [EP/NC] the COUNTABLE SETS are DISTINCT from the UNCOUNTABLE SETS.
- Uncountable set: [EP/NSC] A UNCOUNTABLE SET **is a** SET which cannot number the elements. [EP/NC] the UNCOUNTABLE SETS are DISTINCT from the COUNTABLE SETS.
- Superset: [EP/NSC] A SUPERSET **is a** SET which **contains** all the elements of another SET.
- Subset: [EP/NSC] A SUBSET **is a** SET which has elements **belonging** to ANOTHER UNIT. [SA] The concept of Subset specializes in FULL SUBSET, STRICT SUBSET and COMBINATION, according to the relation: VALUE OF THE CARDINAL.

Relations:

- Has a cardinal: [EP/NSC] TO HAVE A CARDINAL implies to be COUNTABLE. [EP/NC] A CARDINAL counts the number of elements of a SET.
- Contains: [EP/NSC] CONTAINS all the elements of another SET. Implies to be a SUPERSET.

Belonging: [EP/NSC] To have elements BELONGING to another SET. Implies to be a SUBSET.

Definition of the conceptual entities associated to the generic concept "operation" (see Figure 4):

Concepts:

- OPERATION ; **concerns** one or more SETS: [SA] The concept of Operation specializes in COMPLEMENT (NOT), UNION (OR), INTERSECTION (AND) AND CARTESIAN PRODUCT, according to the relation: OPERATION TYPE.
- Complement: [EP/NSC] The COMPLEMENT **is an** OPERATION that **concerns** a SET (A) and a SUBSET (B). The complement of B in A is the set of the elements of A which do not belong to B.
- Union: [EP/NC] The UNION **is an** OPERATION that **concerns** two or several SETS. [EP/NSC] The UNION of two sets A and B is the set of the elements belonging to A or B.
- Intersection: [EP/NC] The INTERSECTION **is an** OPERATION that **concerns** two or several SETS. [EP/NSC] The INTERSECTION of two sets A and B is the set of the elements common to A and B.
- Cartesian product: [EP/NC] The CARTESIAN PRODUCT **is an** OPERATION that **concerns** two SETS A and B. [EP/NSC] the CARTESIAN PRODUCT of A and B is the set of the couples which first element belongs to A and second to B.

Relations:

- Concerns : [EP/NSC] CONCERNS one or more sets. Implies to be an OPERATION.

The following example shows how teaching order information can be expressed:

In order to handle sets, it is necessary to use types of operations, this is specified by the relation "Concerns":
`Concerns`:

- Domain: `complement`.
- Range: `set`.

It is preferable in some situations to use the `Complement` operation to determine the number of combinations of a certain type. For example, if a urn contains 30 red balls and 20 white balls, to solve the question: “how much manners are there to choose five balls with at least one white ball?”, it is better to solve first the question: how much manners are there to choose five balls without white ball. It was also possible to use the `"union"` operation, but with the `Complement` operation, the solution of the problem is easier to find.

With this intention, it is necessary to relate these two concepts `(Set and Complement)` although this relation neither appears in the formal definition of "`complement`" nor in the one of "`set`". For this purpose, we group these two concepts and link an annotation to the group. The annotation specifies the reason why the group is made. The definition of group is in the domain ontology. It is called "`Bag`" and can relate two or more concepts. The link called "`suggestion_annotation`" is defined in the application ontology:

- suggestion_annotation $\subset$ Bag:about $\times$ Annotation:information

Implementation

We have implemented a prototype that allows the management of a learning organizational memory. This prototype has several functionalities related to edition, administration, search and consultation aspects. It uses the Topic Maps formalism to represent information. This formalism is based on three main elements Topic, Association and Occurrence. Since 2001 it has an XML implementation, XTM (TopicMaps.org, 2001). Figure 6 shows how an annotation can be associated to a bag through a suggestion_annotation link, using this formalism and gives an illustration of section *Populating the memory*, line e. “bag_1” and “ann_1” are respectively instances of “Bag” and “Annotation”.

```
<association id= "A-N001" >
    <instanceOf>
          <topicRef xlink:href="#suggestion_annotation" />
    </instanceOf>
    <membre>
          <roleSpec><topicRef xlink:href="#about"/></roleSpec>
          <topicRef xlink:href="#bag_1"/>
    </membre>
    <membre>
          <roleSpec><topicRef xlink:href="#information"/></roleSpec>
          <topicRef xlink:href="#ann_1"/>
    </membre>
</association>
```

Figure 6. Association of an annotation to a suggestion link

Figure 7 shows how to create a link to a resource using the element Occurrence.

```
<topic id="ann_1">
  <occurrence>
    <instanceOf>
      <topicRef xlink:href="#web-page" />
    </instanceOf>
    <resourceRef xlink:href="http://………/statistics/annot1.html"/>
  </occurrence>
</topic>
```

Figure 7. Association of a resource to a topic with an occurrence link

Of course users do not have to know the TM formalism, they access to the memory through an interface. Figure 8 shows a snapshot of the navigation interface. There are two ways to begin to consult the pedagogical content of a course memory. The user can select either one of the available entry points in the left frame, or one of the course notions presented in the right frame. This latter frame initially shows the first specialization level of the application (B31.1 in the example) ontology concepts.

Entry points are chosen by the head of the course. They depend on the learning strategy he wants to enhance. They allow to directly access to a part of the memory. When a user selects an entry point, the associated part of the memory appears in the middle frame. It is in fact a guided navigation through the memory (entry points selected in their presentation order). The user is however free to navigate through the memory by using the left frame if he prefers

In case of a free navigation, information is always presented in the same order in the right frame:

- Path used to reach the selected notion
- Brief text definition of the notion
- Related notions (using the first ontology level)
- Related resources

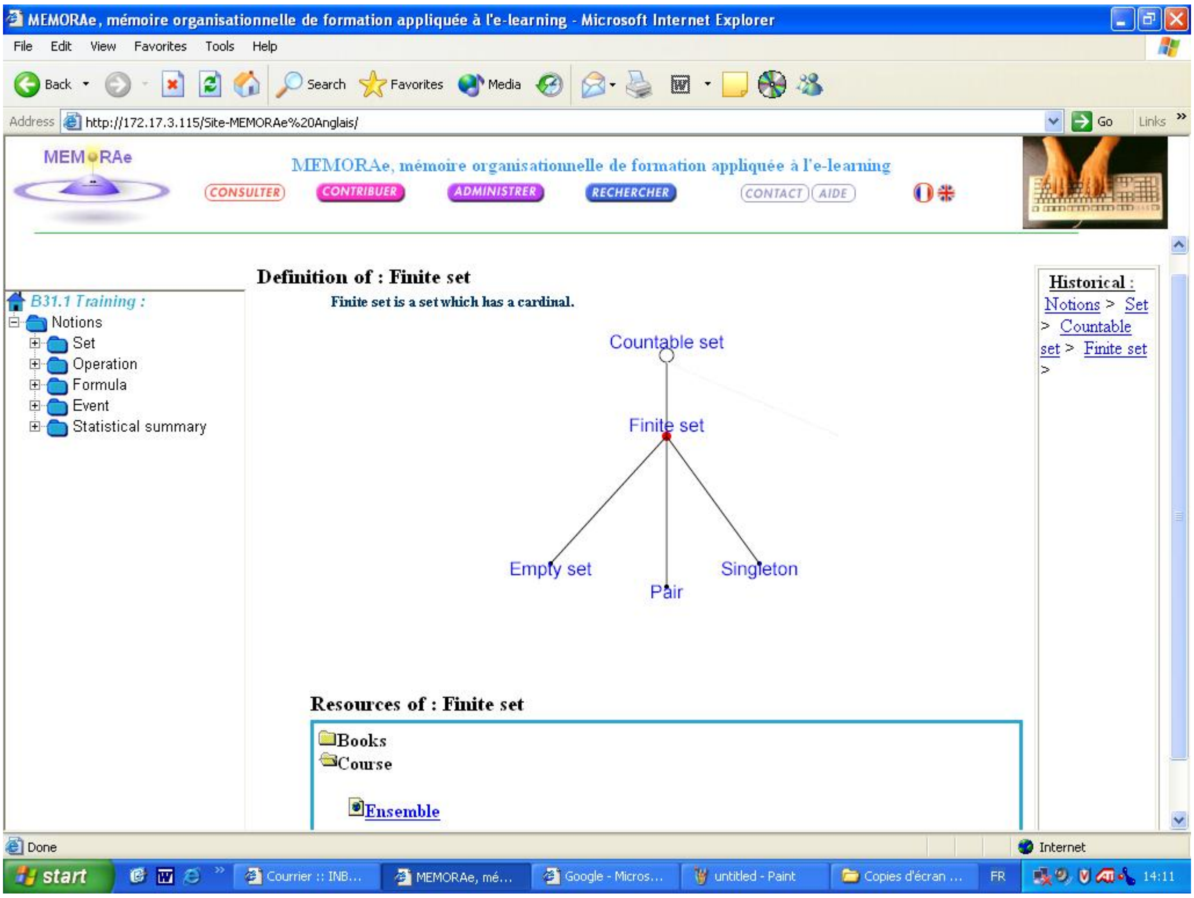

Figure 8. Snapshot of the prototype interface

To access to a resource, user has to select one of its index. Then, he has to click on the term representing the resource in the list of resources associated to this index in order to open a new window containing a description of the selected resource (case of a paper resource) or the resource itself (case of a digital resource).

Related work

Unlike a learning organizational memory proposed in our approach, a Learning Content Management System (LCMS) is an environment where developers can create, store, reuse, manage and deliver learning content from a central object repository, usually a database. LCMS generally work with content that is based on a learning object model (Hall 2001). These systems usually have good search capabilities, allowing developers to find quickly the text or media needed to build training content. They often strive to achieve a separation of content, which is often tagged in XML, from presentation.

From the same author, Learning Objects (LO), also called Reusable Learning Objects, are rather a philosophy for how content can be created and deployed. Learning Objects refer to self-contained chunks of training content that can be assembled with other Learning Objects to create courses and curricula, much the same way a child's Lego blocks are assembled to create all types of structures.

Learning Objects are designed to be used in multiple training contexts, aim at increasing the flexibility of training, and make course updating much easier to manage. When a learning object is updated, the change appears in any course using that learning object.

Within the MEMORAe project, we do not create resources. We choose a set of existing resources with their original format and goal. We index these resources by the means of ontologies (domain and application). These resources are not modified but can be annotated according to the choice of the memory manager. They are retained in their whole, interesting parts can only be highlighted. Authors are free to create resources in their own way. An application ontology has to be built for each course.

Conclusion

In this paper we addressed the problems related to pedagogical resources management for e-learning. To organize the resources in a learning organizational memory, we rely on ontologies. We consider two kinds of ontologies: the first one is generic and concerns the domain of training. The second one is related to the application domain, it is specific to a particular training program. We described our approach for building these ontologies and illustrated it with some examples of the learning memory we designed for two courses of our universities.

This research work is thus situated at the crossroad of three domains: knowledge engineering, pedagogical design and semantic web. The determination of knowledge grains and links between them relates to pedagogical design. The choices of organization, the management of resources in an ontology-based learning organizational memory concerns knowledge engineering. Finally, the choice of the ISO Topic Maps (Park, 2002) standard for representing the memory and allowing its consultation is connected with the semantic web domain.

References

Abel, M.-H., Lenne, D., & Cissé, O. (2002). E-Learning and Organizational Memory. *Paper presented at the International Conference on Artificial Intelligence 2002,* 24-27 June, Las Vegas, USA.

Aussenac-Gilles, N., Biébow, B., & Szulman, S. (2000). Corpus analysis for conceptual modelling. In Aussenac-Gilles, N., Biébow, B. & Szulman, S. (Eds.) *Paper presented at the Workshop on Ontologies and Texts - 12th International Conference on Knowledge Engineering and Knowledge Management*, 2-6 October, Juan-les-Pins, French Riviera.

Bachimont, B. (2000). Engagement sémantique et engagement ontologique: conception et réalisation d'ontologie en ingénierie des connaissances. In J. Charlet, M. Zacklad, G. Kassel et D. Bourigoult. (Eds.) *Ingénierie des connaissances - Evolutions récentes et nouveaux défis, Eyrolles*, 305-323.

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web, *Scientific American*, May 2001, retrieved October 10, 2004 from http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21.

Boullier, D. (2001). Les choix techniques sont des choix pédagogiques : les dimensions multiples d'une expérience de formation à distance. *Sciences et Techniques Educatives, 8* (3/4), 275-299.

Breuker, J., & Muntjewerff, A. (1999). Ontological Modelling for Designing Educational Systems. *Paper presented at the Workshop on Ontologies for Intelligent Educational Systems, Ninth International Conference on Artificial Intelligence in Education (AI-ED'99)*, 19-23 July, Le Mans, France.

Chabert-Ranwez, S. (2000). Composition Automatique de Documents Hypermédia Adaptatifs à partir d’Ontologies et de Requêtes Intentionnelles de l’Utilisateur. *Thèse de l’université de Montpellier II*, France.

Desmoulins, C., & Mille, D. (2002). Pattern-based Annotations on E-books: From Personal to Shared Didactic Content. In M. Milrad, U. Hoppe & Kinshuk (Eds.) *Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE 2002)*, Los Alamitos, CA: IEEE Computer Society, 82-85.

Fernandez, M., Gomez-Pérez, A., & Juristo, N. (1997). METHONTOLOGY: From Ontological Art Towards Ontological Engineering. *Proceedings of the AAAI-97 Spring Symposium Series on Ontological Engineering*, March, Stanford, CA, USA, 33-40.

Fridman, N., & Hafner, C. (1997). The State of the Art in Ontology Design. *AI magazine, 18* (3), 53-74.

Gomez Pérez, A., & Benjamins, V. R. (1999). Overview of Knowledge Sharing and Reuse Components: Ontologies and problem-Solving Methods. *Paper presented at the IJCAI-99 workshop on Ontologies and problem-Solving Methods (KRR5),* 2 August, Stockholm, Sweden.

Gruber, T. (1995). Toward principles for the design of ontologies used for know-ledge sharing. *International Journal of Human Computer Studies, 43* (5/6), 907-928.

Guarino, N., & Giaretta, P. (1995). Ontologies and knowledge bases, towards a terminological clarification. In Mars N. (Ed.) *Towards very large knowledge bases: knowledge building and knowledge sharing*, Amsterdam: IOS Press, 25-32.

Guarino, N., & Welty, C. (2000). Towards a methodology for ontology-based model engineering. *Paper presented at the International Workshop on Model Engineering - 14th European Conference on Object-Oriented Programming*, 13 June, Nice / Sophia Antipolis, France.

Guarino, N. (1992). Concepts, Attributes, and Relations: Some Linguistic and Ontological Criteria for Structuring Knowlege Bases. *Data and Knowledge Engineering, 8*, 249-261.

Hall, B. (2001). *Glossary*, retrieved October 10, 2004 from http://www.brandonhall.com/public/glossary/index.htm.

Kassel, G. (2002). OntoSpec: une méthode de spécification semi-informelle d’ontologies. *Paper presented at the l'ingénierie des connaissances (IC’2002)*, 28-30 May, Rouen, France.

Marshall, C. (1998). Toward an ecology of hypertext annotation. *Paper presented at the Ninth ACM Conference on Hypertext and Hypermedia*, 20-24 June, Pittsburgh, PA, USA.

Park, J. (2002). *XML Topic Maps, Creating and Using Topic Maps for the Web*, Addison-Wesley.

Rabarijaona, A., Dieng, R., Corby, O., & Ouaddari, R. (2000). Building a XML-based Corporate Memory. *IEEE Intelligent Systems*, May-June, 56-64.

TopicMaps.org XTM Authoring Group (2001). *XML Topic Maps (XTM) 1.0: TopicMaps.org Specification*, retrieved October 10, 2004 from http://www.topicmaps.org/xtm/index.html.

Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, Methods and Applications. *Knowledge. Engineering Review, 11* (2), 93-155.

Van Heijst, G., Schreiber, A., & Wielinga, B. (1997). Using Explicit Ontologies in KBS Development. *International Journal of Human-Computer Studies, 46*, 183-298.