BUILDING PERSONAL KNOWLEDGE THROUGH EXCHANGING KNOWLEDGE CHAINS

Juliana Lucas de Rezende, Rafael Leonardo Siqueira da Silva
COPPE/UFRJ - Computer Science Department, Graduate School of Engineering, Federal University of Rio de Janeiro
PO Box 68.513, ZIP Code 21.945-970, Cidade Universitária – Ilha do Fundão, Rio de Janeiro, RJ, Brazil
ju_lucas@yahoo.com.br, rafa_leo@cos.ufrj.br

Jano Moreira de Souza
UFRJ - Institute of Mathematics, Federal University of Rio de Janeiro and
COPPE/UFRJ - Computer Science Department, Graduate School of Engineering, Federal University of Rio de Janeiro
PO Box 68.513, ZIP Code 21.945-970, Cidade Universitária – Ilha do Fundão, Rio de Janeiro, RJ, Brazil
jano@cos.ufrj.br

Milton Ramirez
UFRJ - Institute of Mathematics, Federal University of Rio de Janeiro
PO Box 68.513, ZIP Code 21.945-970, Cidade Universitária – Ilha do Fundão, Rio de Janeiro, RJ, Brazil
milton@cos.ufrj.br

ABSTRACT

To complement the learning process, this paper presents a process for building personal knowledge through exchanging knowledge chains in learning communities. The process differential is the addition of "how to use" the available knowledge to its authors (who), localization (where), and content (what), that are commonly used. The hypothesis is that the apprentice will achieve a reduction in the time dedicated to studying a specific subject, as well as obtain more comprehensive knowledge of the subject studied. The experimental use of the prototype developed showed evidence that the hypothesis can be confirmed.

KEYWORDS

Knowledge Chains, Constructivism, Computer-Supported Cooperative Work, Knowledge Management, Peer-to-Peer Architecture, Knowledge Object.

1. INTRODUCTION

Today, people need to acquire new knowledge faster and in a much greater volume than in the past. To complement the learning process, there are communities of practice focused on learning, i.e., learning communities. These communities act as both, a method to complement teaching in the traditional classroom and to acquire knowledge in evolution. Pawlowski at al. (2000) defined a learning community as being an informal group of individuals engaged in a common interest, which is, in this case, the improvement of the apprentice's performance, using computer networks. One of the principles of Wenger et al. (2002) for cultivating communities of practice is the sharing of knowledge to improve personal knowledge. Another issue toward making a successful community should be intense communication among members. Finally, a community should assist the members in building up their personal knowledge. (Tornaghi, 2004)

The aim of this work is to develop a system that supports learning communities and increases the learning process, promoting knowledge building, dissemination, and exchange, based on a knowledge chains approach. So, in this paper, a proposal of the system architecture model will be presented.

To make efficient knowledge building, dissemination, and exchange, it is necessary to do more than store data. It is necessary to manage different kinds of information, such as the resource description, the authors, keywords, and others. This kind of information about the information itself is called meta-information, and its scope is to be machine understandable, so that it can be used by software agents to easily retrieve resources from a huge pool of resources, like the Internet, business data warehouses, and other kinds of data storage. (Oliveira, 2004)

New technologies, like high-speed networks, allows real-time cooperative work. (Fluckiger, 1995) In this sense, the peer-to-peer (P2P) approach can greatly contribute to cooperative work. For instance, P2P architecture permits direct interaction between peers, which promotes more dynamic application. P2P applications are naturally not centralized, and this fact allows applications to form small groups of cooperation without the presence of a central server, which could be stressed by the number of connections. In other words, the P2P architecture offers a more robust and fault-resilient environment for cooperative work. (Chawathe, 2003)

The remainder of this paper is organized as follows. The next section presents important concepts that are the basis for this work. Section 3 discusses a number of theoretical knowledge chain aspects. In section 4, we present the proposed idea and the prototype developed. Future work and the conclusion are shown in section 5.

2. BACKGROUND

2.1 Constructivism

Constructivism is a psychological theory initially developed by Jean Piaget. In it, true knowledge is defined as stemming from a personal elaboration (or construction) which results from an internal thought process. During this process, the individual coordinates different knowledge constructs, giving them meaning, organizing them, and associating them to other knowledge previously acquired. This process is intransferable, as no one person can accomplish it for another, since knowledge is constructed from an individual's particular need. (Sastre, 1997)

A method of study based on constructivism has as its basis the idea that both the act of teaching and the act of learning mean building new knowledge based on experiences and knowledge previously acquired. In this way, constructivism stimulates the learner's thought process to reconstruct previously acquired knowledge, instead of forcing him to passively take in the content passed on by the teacher. (Sastre, 1997)

2.2 Knowledge Management (KM)

There are many definitions for the concept of KM. Some, like Despres and Chauvel (1999), have come to the conclusion that, although KM is important, it is intellectually evasive. Nonaka (1994) defines KM as coordination, transference, and transformation of knowledge. The aim of KM is to identify the knowledge in the collective memory and facilitate communication between people who create knowledge and people who need it.

According to Nonaka (1994) and Sveiby (1996), the nature of knowledge can be divided into tacit (non-codifiable knowledge) and explicit (codifiable knowledge). The basic idea is that both tacit knowledge and explicit knowledge are connected by processes that transform tacit knowledge into explicit knowledge and vice versa.

2.3 Computer Supported Cooperative Work (CSCW)

According to Grudin (1994), the concept of CSCW was introduced in the mid-1980s, in a conference that brought together researchers in many different areas. The main interest of the gathering was studying how people carried out their work activities and how the existing technology could help them. From that moment on, there was growing interest in the subject in the areas of workgroup support systems, management information systems, education, social sciences, and other disciplines. This and other definitions along the same lines are subject to criticism by some segments in the area of CSCW research. However, they manage to reflect a good amount of consensus in this research area.

2.4 Peer-to-Peer Architecture and COPPEER

During the 1990s, computer networks promoted an explosion of tools geared towards communication operations and, currently, it is possible to transmit a significant amount of data over the Net. Nowadays, the client-server paradigm is the dominant one in software architectures, but, in recent years, the interest in peer-to-peer (P2P) computing has grown. P2P architecture potentially offers many possibilities, not totally explored, which make it a great competitor to client-server architecture. P2P architecture is characterized by the independence of the network nodes; it doesn't have the presence of a central server. Each node is independent and self-sufficient. A node can play a special role inside the network; however, it can be substituted by any other node in case of connection failure. In this sense, P2P architecture is extremely modular and fault-tolerant. (Chawathe, 2003)

New technologies have been created to support these distributed systems and help build them, but developers still have to design new applications almost from scratch, duplicating much functionality each time. The current P2P framework research has focused especially in the search problem, which has proved to be a bottleneck in current P2P applications. Both centralized and fully distributed approaches, suffer in terms of scalability. Hybrid techniques have been investigated and resulted in proposals like JXTA's architecture (2004). Distributed search, therefore, is one of the main goals of P2P architectures, being actually critical to its success.

COPPEER is a framework for creating very flexible collaborative P2P applications. The idea is to build applications as sets of services, controlled by a micro-kernel. In this fashion, a user would have different applications (file sharing, messaging, conferencing, commerce, etc.) within the same supporting environment, and new services could easily be added as components. Most P2P mechanisms in COPPEER are provided by JXTA modules; therefore, the framework does not compete or substitute JXTA, but rather creates a higher level of abstraction to facilitate the development of P2P and collaborative applications. (Xexeo, 2004)

2.5 Knowledge Object (KO)

There are serious questions concerning whether standards provide mechanisms to supply knowledge creation and knowledge reuse rather than merely to describe information. Some issues are not available, such as reusing content at all levels, permitting the combination (manual or automatic) of pieces of knowledge to create new knowledge, etc. There are some proposals in the literature referred to as "instructional object", "educational object", "learning object", "knowledge object", "intelligent object", and other terms.

A more precise approach to knowledge representation, organization, and use for learning and collaborative purposes is the knowledge objects concept defined by Merrill (1997). A knowledge object is a framework for identifying necessary knowledge components, which is a precise way to describe the subject matter content or knowledge to be taught. Knowledge objects intend to describe every kind of knowledge, facilitating search, reuse, management, and exchange. (Oliveira, 2004)

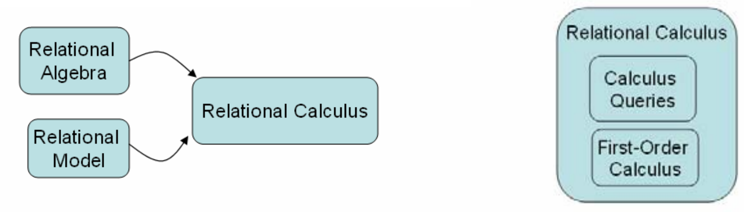
3. KNOWLEDGE CHAINS (KC)

To understand what KCs are, one must be familiar with some concepts related to knowledge structure and organization. There are many ways to organize knowledge. Some of them are presented as follows:

- Outline: the main ideas or facts about something, without the details; a sketchy summary of the main points of an argument or theory; a step-by-step guide, usually constructed for your own benefit, showing how you plan to write a particular manuscript
- Summary: a short statement that gives the main information about something, without giving all the details; brief abstract; a brief restatement of the main points of a report
- Curriculum: the subjects taught by a school; is the series of courses in which students are introduced to and needed for a graduate; a structured plan of instruction that details what students are to know, how they are to learn it, what the teacher's role is, and the context in which learning and teaching will take place. The learning plan is generally organized as a sequenced combination of modules

- Prerequisite Chain: chains represented by charts that show the prerequisites between offered courses; they show the order in which courses must be taken as well; they refer to the school curriculum

The KC definition is based on knowledge structure and organization. There are two ways to organize knowledge. The first is similar to the way presented in the prerequisite chain. There is a relationship between the current KU and all its prerequisites and between the current KU and all its successors. A knowledge prerequisite is any KU necessary to learn the current KU, and a knowledge successor is any KU that can be learned after the current KU learning. Figure 1.a presents an example where Relational Algebra and Model are prerequisites of Relational Calculus, so Relational Calculus is their successor. The other way to organize knowledge is by composition. When a KU is formed by composition of other KUs, it can be represented like in Figure 1.b. In this example, Relational Calculus is composed of Calculus Queries and First-Order Calculus.



a. Knowledge Chain b. Knowledge Composition

Figure 1. Knowledge Organization

A KC is made up of a header (which contains basic information related to the chain) and a knowledge units multlist. So, to define a KC, it is necessary to define a knowledge unit (KU). Conceptually, knowledge can be decomposed into smaller knowledge. Such decomposition may occur recursively. For simplification effects, we will consider that there is a basic unit which can be represented as a KU. A KU is a structure formed by an attribute set. The attributes are grouped into categories, as presented bellow:

- General – general information about the KU as a whole;
- Life Cycle – history, current state, and contributors;
- Rights – intellectual property rights and conditions of use;
- Relation – the relationship between knowledge resources;
- Classification – the KU in relation to a classification system;
- Annotation – comments and evaluations of the KUs and their creators.

Each one of these categories attempts to characterize the KU by a separate aspect, and these categories represent a group of data elements which may or may not contain sub-elements.

The “general” category contains general information about a KU, such as: name, description, keywords, author, creation date, last use date, etc. In the “life cycle” category, history info is stored through current state, contributors, etc. Contributors are those who build a new KU by changing an existing one. A KU can have three different states (question, solution formula, learned) in agreement with KC apprentice context.

The “rights” category contains access information. A KU can have three different access types (public, private, referenced). When a KU is created, the author decides if its access will be public or private; referenced access occurs when an existing KU is incorporated to the apprentice context.

References, associated files, prerequisites, successors and children are included in the “relational” category. Links, books, journals, and papers are simple reference examples; a KU may also be related, and it occurs automatically (a related KU is the existing one modified to build a new one) to guarantee the copyrights. It is possible to associate any type files to a KU, and these files are always transmitted with the KU. Children list contains all KUs stored in the composed KU.

The “classification” category contains attributes that classify the KU, such as time (estimated time needed to learn the KU), cost (estimated price of the knowledge referenced by the KU), quality (estimated quality of learning), category (category in which the KU is inserted), etc.

The “annotation” category contains comments on KUs and their creators.

Figure 2 presents an instance of a KU based on KO’s aspects.

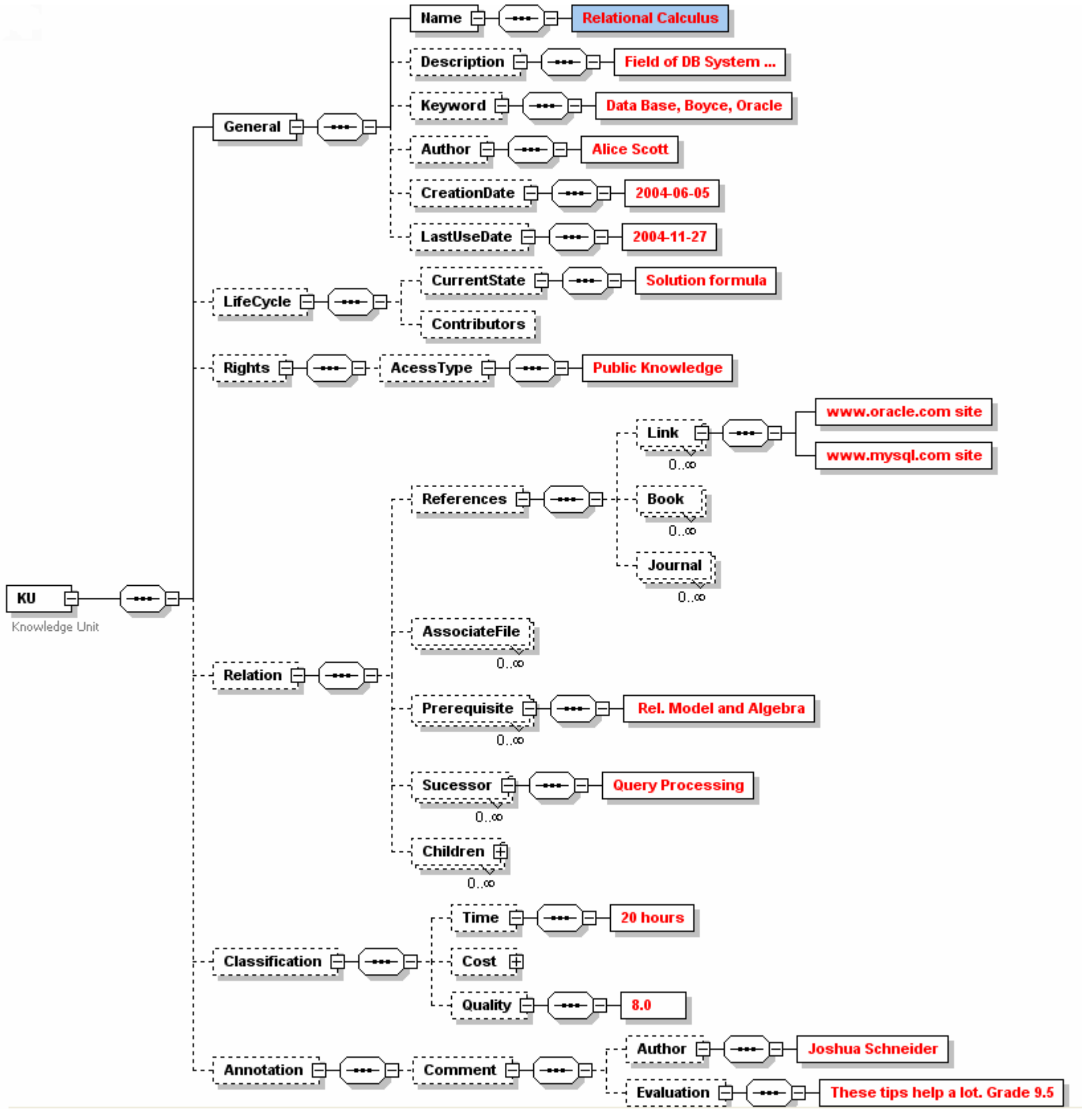


Figure 2. Instance of Knowledge Unit

4. BUILDING PERSONAL KNOWLEDGE CHAINS

The main target of this work is to build a process that promotes knowledge building, dissemination, and exchange. For this, the proposal is to develop an application, called Knowledge Chains Editor (KCE), to build personal KCs. Figure 3 presents the KCE architecture.

The members of the community should build their personal knowledge based on the knowledge already present in the community, and, to make that possible, the process developed bases itself on the paradigm of constructivism. In the proposed architecture, like in constructivism, the two main characteristics for knowledge building are that the building should take place through the active involvement of the apprentice; and the previously existing knowledge of the participants plays an important role in the process of building new knowledge.

According to Davenport (200), the knowledge process can be divided into three phases: knowledge generation, codification, and transfer. And the concept of transfer into two phases: transmission and absorption. The system proposed here intends to become a facilitator of knowledge transfer, focusing on the transmission phase. The use of this system does not guarantee the transfer of knowledge, but it can help the meta-knowledge - which, in this context, would be the KU - reach those who need it.

According to CSCW theories, it is believed that combination and reuse of explicit knowledge are encouraged when a cooperative environment is proposed. For this reason, our proposal is based on the concepts of CSCW.

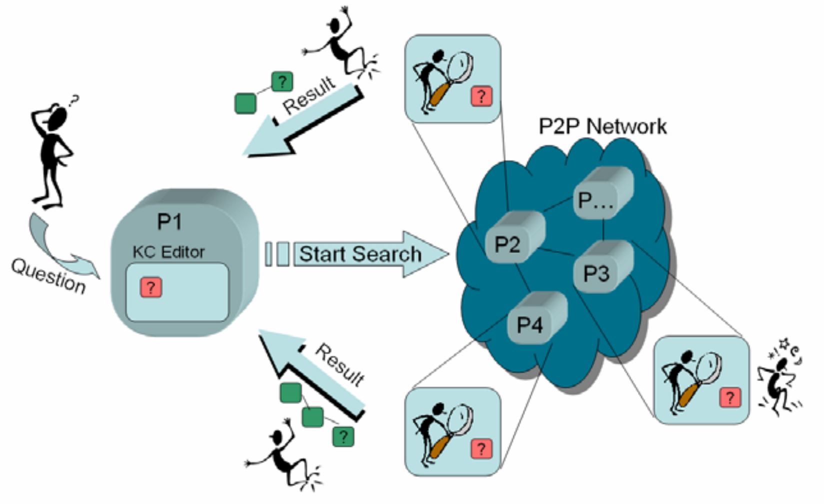


Figure 3. Knowledge Chains Editor Architecture

To build his KC, the apprentice can use the KCE. For each KU created, the apprentice needs to fill out all the mandatory information. In case of questioning, the apprentice must create a KU whose state is "question". At this moment, the system starts the KU search. It sends messages to other peers and waits for an answer. Each peer is responsible for the internal search. The internal search consists of verifying if there are any KUs similar to the one in the search. All KUs found are returned to the solicitor. For each KU found, it is necessary to return the KC where the searched-for KU is inserted. In cases where the peer can't find anything, it does nothing.

The KCE presents to the apprentice all KCs found. At this moment, the apprentice must choose the KC that better serves his needs. The KCE will help him choose the best alternative. This functionality is based on the capacity to point out better alternatives developed by the Bill of Experiments system (Cardoso, 2002), a collaborative system for explicitation, reuse, and planning of scientific workflows capable of pointing out better alternatives to carry out scientific experiments, based on cost criteria, deadline attainment, and quality. The module developed to optimize the learning strategies was named Bill of Learning (BOL). It represents the KCs as logical expressions and uses inference rules (written in Prolog) to decide which is the best alternative according to the apprentice's needs. The choice of the best alternative may be based on time, cost, quality, or best cost/benefit relationship between those three characteristics. The apprentice himself decides which characteristic should be taken into consideration. After the selection of the best alternative, the apprentice can study using de KC created. For each learned KU, the apprentice must update its state (learned). The search will be finished when the apprentice selects any returned KU into his personal chain, when the stipulated time is over, or when the apprentice decides to end the search.

An important question is how to compare two KUs and how to say they are similar. The use of ontologies can make that easier. Another important point is the granularity in which a KU will be represented. Defining the level of detail used to represent knowledge is particularly difficult. Time and quality can be used to facilitate this definition, for example, the greater the time, the smaller the granularity. So, the calculation must be done to check whether two apparently similar KUs have the same granularity.

4.1 Knowledge Chains Editor (KCE)

The KCE is part of a more global concept, the Apprentice Profile. Profile is a whole description of a person that contains not only information about the knowledge, but also about the preferences, issues, and interesting aspects of a person. The KCE allows us to change the KCs and get more information from the source of knowledge so we can make a better choice when choosing. For example, a KC about Calculus available in a Math professor profile is more likely to be right than one available in a Biology professor profile. Knowing the source of the KC may be very useful, but we need some special structure to represent the profile without losing the semantics. The solution adopted is based on KO representation proposed by Oliveira (2004). The main drawback of this approach is the total work needed to adapt the KO to our needs; however, the benefits brought by it are much greater.

From now on, we will examine the KCE more in depth. Its main feature is to promote the exchange of KCs between the nodes of the network. The exchange is possible thanks to the P2P infrastructure, which supports the application. The KCE was developed to work inside COPPEER, and because this, the application has many facilitating features, such as concurrence control, a search mechanism and awareness system (which enable to know when a peer fails). The P2P architecture enables the KCE to be more dynamic, resilient, and fault-tolerant. The absence of a server permits the KCE to work simply connected to other KCEs and make queries about the desired profile. Even if some profiles are not available at the time of the query, it is possible to get other useful profiles, which would be impossible by using a server.

Figure 4 presents the KCE Interface. The main window is made up of three main fields. On the left, there is a tree representation of the profile being presented. On the right, there is a graph representation of the same profile. The graph makes the dependency between the researcher's KUs clear. At the top, there is a box of "Details", which shows, in detail, the information related to each KU.

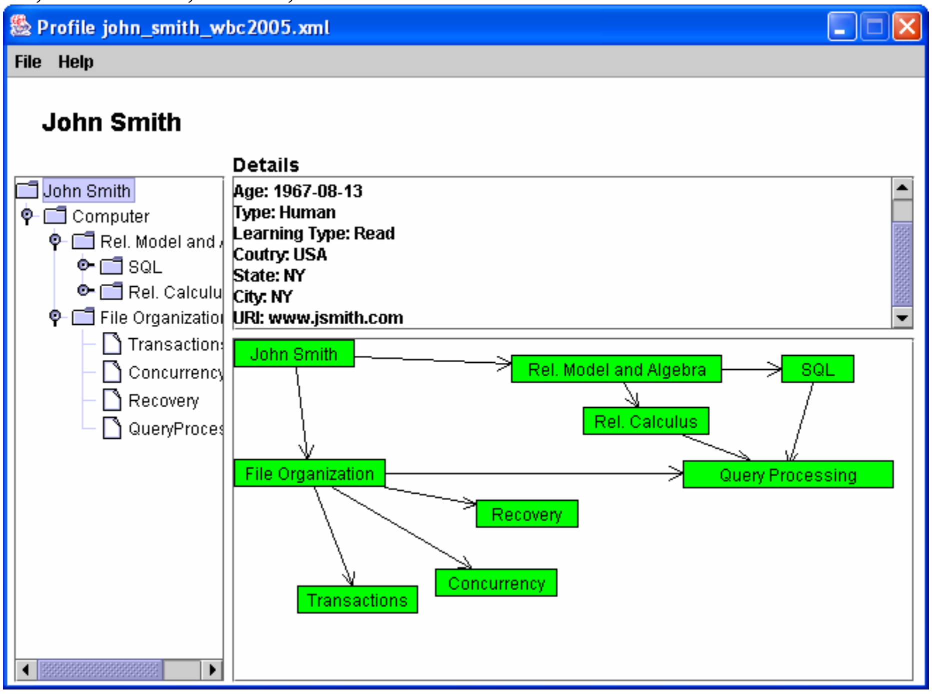


Figure 4. Knowledge Chains Editor Interface

The Relational Calculus KU is the graphic representation of the instance presented earlier, in Figure 2.

5. CONCLUSION AND FUTURE WORKS

The availability of P2P technologies and the growing number of learning communities which communicate online makes it possible to exchange, select, and use chains of explicit knowledge as a strategy for creating personal knowledge. Today, we have the WWW (who, what, where) triad, where "who" are the people who have the knowledge, "what" is the knowledge itself, and "where" is its location - in our case, the peer in which it is located. This paper hopes to add "how to use" the available knowledge to the existing triad.

The creation of a KU of question type is obviously motivated by the apprentice's need to obtain that knowledge. We have considered that there are two motivating factors for the creation of KCs to be made available. The first would be a matter of recognition by the communities, since each KU created has a registered author. The second would be the case where the professor makes them available "as a job", with the intention of guiding his students' studies.

The experimental use of the KCE showed evidence that, when used by an apprentice in building a personal KC, the hypothesis that he would achieve a reduction in the time dedicated to studying a specific subject, as well as obtain more comprehensive knowledge of the subject studied was confirmed. In order to evaluate the reach of the KCE's objective, experiments aimed at obtaining qualitative and quantitative data that would make possible the verification of the hypothesis under consideration must be carried out.

It is necessary to point out that it is not the goal of this work to ensure that the apprentice has assimilated everything in his chain of knowledge. Our goal is to offer support for the apprentice's learning process, according to his priorities, which may be access to the chain with the best cost/benefic relationship between time and quality, and others.

Due to the fact that this work is still in progress, many future projects are expected to ensue. We present a few of those here: enriching the profile with other editors such as a project editor; adding new types of searches, such as a search for successors, which would solve problems like: "What can one learn starting from what he already know?"; adding a recommendation system module based on the successors search; applying that idea to a mobile environment, with the aim of taking advantage of its characteristics (ubiquity).

ACKNOWLEDGEMENT

This work was partially supported by CAPES and CNPq.

REFERENCES

Cardoso, L. F., Souza, J. M., Marques, C., 2002. *A Collaborative Approach to the Reuse of Scientific Experiments in the Bill of Experiments. Proceedings of 7th International Conference on CSCW in Design.* Rio de Janeiro, Brazil.

Chawathe, Y., et al, 2003. *Making Gnutella-like P2P Systems Scalable. Proceedings of ACM SIGCOMM*. University of Karlsrehe, Germany.

Davenport, T., Prusak, L., 2000. *Working Knowledge – Knowledge Transfer.* Harvard Business Review, pp. 88-106.

Despres, C., Chauvel, D., 1999. Knowledge management. *Journal of Knowledge Management*, Vol. 3, No. 2, pp 110-120.

Fluckiger, F., 1995. *Understanding Networked Multimedia, Applications and Technology.* Prentice Hall, Hemel Hemstead, England.

Grudin, J., 1994. Computer-Supported Cooperative Work: History and Focus. *In IEEE Innovative Technology for Computing Professionals,* Vol. 27, No. 5, pp 19-26.

JXTA Project (2004). Available online at: http://www.jxta.org.

Nonaka, I., 1994. The Dynamic Theory of Organization Knowledge Creation. *Journal of Organization Science*, Vol. 5, No. 1, pp 14-37.

Oliveira, J.; Souza, J.M., 2004. *Improving Knowledge Sharing through Knowledge Objects Representation*. *Proceedings of the 4th International Conference on Knowledge Management* (I-KNOW '04), Graz, Austria.

Pawlowski, S., et al, 2000. *Supporting shared information systems: boundary objects, communities, and brokering. Proceedings of the 21th International Conference on Information Systems.* Brisbane, Australia, pp. 329-338.

Sastre, G., et al, 1997. *Temas transversais em educação – bases para uma formação integral.* Ed. Ática, São Paulo, Brazil.

Sveiby, K., 1996. Transfer of Knowledge and the Information Processing Professions. *European Management Journal,* Vol. 14, No. 4, pp. 379-388.

Tornaghi, A., Vivacqua, A., Souza, J.M., 2004. *Creating Educator Communities. Proceedings of IADIS International Conference on Web Based Commuties.* Lisbon, Portugal.

Wenger, E., et al, 2002. *Cultivating Communities of Practice: A guide to Managing Knowledge*. Harvard Business School Press.

Xexeo, G., Vivacqua, A., Souza, J.M., et al, 2004. *Peer-to-Peer Collaborative Editing of Ontologies. Proceedings of 8th International Conference on CSCW in Design.* Xiamen, China.