Using Ontological Engineering to Overcome Common AI-ED Problems

Riichiro Mizoguchi, Jacqueline Bourdeau

International Journal of Artificial Intelligence in Education (2000), 11, 107-121

Using Ontological Engineering to Overcome Common AI-ED Problems

Riichiro Mizoguchi *ISIR, Osaka University, 8-1 Mihogaoka, Ibaraki, Osaka, 567-0047 Japan, email: miz@ei.sanken.osaka-u.ac.jp*

Jacqueline Bourdeau *Université du Québec à Chicoutimi, 555 Boulevard de l'Université, Chicoutimi, G7H 2B1 Canada, email: jbourdea@uqac.uquebec.ca*

Abstract. This paper discusses long-term prospects of AI-ED research with the aim of giving a clear view of what we need for further promotion of the research from both the AI and ED points of view. An analysis of the current status of AI-ED research is done in the light of intelligence, conceptualization, standardization and theory-awareness. Following this, an ontology-based architecture with appropriate ontologies is proposed. Ontological engineering of IS/ID is next discussed followed by a road map towards an ontology-aware authoring system. Heuristic design patterns and XML-based documentation are also discussed.

1. INTRODUCTION

Among AI-ED research done to date, several paradigms such as CAI, ICAI, Micro-world, ITS, ILE, and CSCL have been proposed and many systems have been built within each paradigm. Additionally, innovative computer technologies such as hyper-media, virtual reality, internet, WWW have significantly affected the AI-ED community in general. We really have learned a lot from our experiences and we may say the results are very fruitful. Moreover, we still need to identify promising directions to which effort should be devoted for further progress of AI-ED research.

AI-ED research consists of the following three major research areas:

1. Artificial intelligence (AI)
2. Instructional design science (IS)
3. Learning science (LS)

Some AI and IS technologies have successfully been introduced into the AI-ED community bearing fruitful results in building various Intelligent Instructional Systems (IIS). However, to date, the interaction between AI-ED and AI communities and AI-ED and IS communities have not been very active. The authors believe that promotion of more active interactions between AI-ED and AI&IS communities would be beneficial for the future AI-ED research.

This paper discusses long-term prospects of AI-ED research with the aim of giving a clear view of what we need for further promotion of the research from AI and ED points of view. The main topic here includes how to introduce ontological engineering (Mizoguchi, Sinitsa and Ikeda, 1996a; Mizoguchi, 1998) into AI-ED research. The next section enumerates the shortcomings that current AI-ED research suffers from and discusses the four underlying concepts of intelligence, conceptualization, standardization and theory-awareness. In section 3, we review what is happening in the knowledge-based systems community to learn how ontological engineering plays roles critical to overcoming the problems common to IISs. On the basis of these observations, we discuss ontological engineering of Instructional design:ID knowledge as one of the promising directions to build an ontology-aware authoring environment. Following this, we present a road map towards such an environment supported by IS and ID. Finally, some other interesting topics are discussed followed by concluding remarks.

2. ANALYSIS OF CURRENT STATE OF THE ART OF AI-ED RESEARCH

Let us first enumerate drawbacks of current IISs from AI and ED point of views.

1. There is a deep conceptual gap between authoring systems and authors.
2. Authoring tools are neither intelligent nor particularly user-friendly.
3. Building an IIS requires a lot of work because it is always built from scratch.
4. Knowledge and components embedded in IISs are rarely sharable or reusable.
5. It is not easy to make sharable specifications of functionalities of components in IISs.
6. It is not easy to compare or cross-assess existing systems.
7. Communication amongst agents and modules in IISs is neither fluent nor principled.
8. Many of the IISs are ignorant of the research results of IS/LS
9. The authoring process is not principled.
10. There is a gap between instructional planning for domain knowledge organization and tutoring strategy for dynamic adaptation of the IIS behavior.

All these issues are content-related ones. In other words, neither inference techniques nor beautiful theoretical formalism can contribute to an improvement of the situation. The authors' claim that what we need to overcome these drawbacks is ontology-based architecture and appropriate ontologies, that is, the introduction of ontological engineering. Now, let us analyze the situation in more detail to see the justifications of how ontological engineering can make a useful contribution.

2.1 Intelligence

Adaptivity is at the heart of intelligent systems. It comes from the declarative representation of what the system knows about the world it is in. In IIS cases, the world consists of a learner and the system itself. So such a system behaves adaptively with respect to the learner's state of understanding. A learner model, which is a representation of system's knowledge about the learner, serves as the source of intelligence. The system can investigate the learner model to adapt its behavior to the learner. In this sense, the model should be represented declaratively in order for the system to update and interpret.

What about authoring systems? Do they have such a model or declarative representation of what they know? Unfortunately, the answer is No. This is one of the major reasons why authoring systems are not intelligent. Intelligence of authoring systems is not an issue specific to authoring system research but a general issue common to most AI-ED research, since it is deeply related to knowledge of how to build IISs.

As discussed above, the source of intelligence of the systems is declarative descriptions of what they know. They can dynamically inspect these declarative descriptions in order to adapt their behavior to the circumstances under which they are operating. We could say the intelligent systems know what they know and what they are doing.

The next issue is what knowledge authoring systems should possess. Conventional authoring can be viewed as a kind of knowledge acquisition (KA) from teachers/ instructors/trainers. This view gives us a few constructive suggestions to improve the performance of authoring systems. We could learn from the research of knowledge acquisition in knowledge-based systems community where KA from domain experts has been the bottleneck of expert system building. One of the major causes of the difficulty is due to the heavy dependence on heuristic knowledge, which is hard to acquire, to manipulate and justify. This is why model-based approaches are incorporated in modern knowledge-based systems which rely mainly on domain theory and model of the target system instead of heuristics.

In our case, analogously, a "Model-based approach" is strongly needed to make IIS building more scientific and principled. We need to depart from the "heuristics-based approach" and employ a new technology, that is, ID knowledge to enable the development process to follow instructional design decisions based on principled and theory-based knowledge.

2.2 Conceptualization

Conceptualization is an AI term and consists of a set of objects existing in the target world and the relationships between them. It forms the basis on which many systems' performance heavily depend. One of the major difficulties common to most existing systems is the lack of an explicit representation of the conceptualization each systems is based on. Taking an expert system as an example, it has a knowledge base in which it declaratively represents what it knows about problem solving in the task domain. The knowledge base can be a source of intelligence used to solve problems. Once a user tries to modify the knowledge base or to reuse some portion of an existing knowledge base developed by other people, however, he/she immediately has serious difficulty in doing so. This is due to the fact that the conceptualization of the knowledge base as well as the underlying assumptions are implicit. Even worse, the same terms may have different meanings to each user. Such information has rarely been represented explicitly, which has been a serious cause of many drawbacks that current knowledge-based systems suffer from.

The same applies to IISs. Few IISs have an explicit representation of their conceptualization. While they know about the understanding state of the learner and have tutoring knowledge, etc., they do not know any concept of which their knowledge is composed. In other words, most of the concepts, as well as their primitive actions are hard-wired. We could say that such systems are illiterate because they do not understand the basic concepts. Therefore, such systems will never be able to communicate with users (authors) in terms of fundamental knowledge.

An authoring system should be aware of this conceptualization of an IIS because it creates an IIS by manipulating concepts according to the design rationale employed.

2.3 Standardization

Needless to say, industries have attained today's high productivity thanks to the standardization of components, for example, bolts and nuts. It is a pity that we have no such standardized components in the AI-ED community. In order to model target objects, such components would help a lot and facilitate model-based problem solving. Standardization of components does not necessarily imply standardization of knowledge in general. We are not claiming that all of the knowledge should be standardized. Using standardized basic components, one can easily design one's own knowledge by configuring them, as is supported by the current engineering production.

Standardization is mainly for providing us with a common vocabulary for understanding what has been done to date with less ambiguity. It never implies any restriction to the exploration of future research activities. The main reason why humans can communicate with each other is that we have a common platform of meaning on which we can rely and common concepts in terms of which we can express our ideas. We can say that standardization, at least an attempt at standardization in order to find not only what we can standardize but also what we cannot standardize, is crucial to the further success of AI-ED research.

However, one may say that AI-ED research would be premature in establishing a standard. If "standardization" is a bit too strong, we could use the term "shared vocabulary". It sounds much more gentle and acceptable. Specification of functional components should be described in terms of common vocabulary. The problem, however, is that the terminology used by teachers, authors, and developers are different from each other. As is discussed above, implemented systems do not understand either of the vocabularies. In short, none of the four participants, three humans and one computer, share a common vocabulary. This has caused a lot of misunderstanding.

This is even the case among human participants. Furthermore, when they start discussion on comparison between several IISs of different domains, it is not easy for them to properly perform the comparison because of the different terminologies used in the respective systems.

The same applies to communication between component modules in IISs. Because no common functional specifications or vocabulary are available, the components can not communicate with each other properly. This is one of the main factors which prevents their reuse.

2.4 Theory-awareness

In general, expertise is composed of heuristics and domain theories in general. Once it is extracted and is at hand, heuristic knowledge is easy to implement and resulting systems usually works well. Many knowledge-based systems have been built employing heuristic knowledge. One of the shortcomings of heuristic approach is that it is not principled and it ignores existing theories. Another way of building a knowledge-based system is to use domain theories which are, in general, objective and convincing. People can easily accept such systems that are based on theories.

The issue here, however, is that all of the theories in many of the theory-based systems are built-in in the procedures. The developer, not the system, knows the theory. Such a system cannot be said to be "theory-aware". Authoring systems, which are a kind of meta-system in the sense that they generate IISs, need to be "theory-aware" to be intelligent. The rich accumulation of instructional and learning theories should be used to make authoring systems knowledgeable. Declarative representation of those theories enables them to be called "theory-aware".

2.5 Summary

Making systems intelligent requires a declarative representation of what they know. Conceptualization should be explicit to make authoring systems literate and intelligent, standardization or shared vocabulary will facilitate the reusability of components and enable sharable specification of them and theory-awareness makes authoring systems knowledgeable. The next problem is how to find a solution which satisfies all of these requirements. Knowledge systems research in AI suggests that ontological engineering could provide us with a solution. Let us investigate what has been happening in the knowledge processing community.

3. KNOWLEDGE AND ONTOLOGICAL ENGINEERING

3.1 Knowledge modelling and ontology

In the expert system community, the knowledge principle, "The power exists in the knowledge", proposed by Feigenbaum has been accepted and carried out with a deep appreciation, since it is to the point in the sense that the importance of the accumulation of knowledge is larger than that of formal reasoning and logic in making expert systems work. This has been proved by the success of expert system development and a lot of research activities have been carried out under the flag of "knowledge engineering". After an initial success, however, people realized serious difficulties in knowledge base technology, that is, the expense of building each knowledge base and the low reusability of knowledge bases. They tried to deal with heuristic knowledge using simple rule base technology. They did not have any sophisticated methodologies or theories for eliciting knowledge from the knowledge sources or transforming, organizing, and translating the domain knowledge to enable the computer to utilize it.

Knowledge modelling, which is a substitute for rule base technology, is a new technology for overcoming such difficulties. It has made it easier to elicit expertise, to organize such expertise into a computational structure, and to build knowledge bases. Examples include the KADS project in Europe (Breuker and de Velde, 1994), the PROTEGE project in USA (Puerta and Musen, 1992) and the MULTIS project in Japan (Mizoguchi, Tijerino, and Ikeda, 1992; Mizoguchi, Vanwelkenhuysen, and Ikeda, 1995). All these technologies originate from the concept of Generic tasks (Chandrasekaran, 1986) and heuristic classification (Clancey, 1985). The underlying idea is to find domain-independent activities which specify the roles that the domain objects play in the problem solving process. The latest knowledge modeling research introduces the idea of ontology. Roughly speaking, an ontology consists of task ontology, which specifies the problem solving architecture of knowledge-based systems, and domain ontology, which specifies the domain knowledge. The concept of task ontology has been proposed by one of the authors and it serves as a theory of vocabulary/concepts used as building blocks for the modeling problem solving structure (Mizoguchi, Vanwelkenhuysen, and Ikeda, 1995; Mizoguchi, Sinitsa, and Ikeda, 1996a; Chandrasekaran, Josephson and Benjamins, 1999).

Task ontology provides us with an effective methodology and vocabulary for both analyzing and synthesizing knowledge-based systems. It is a system of terms/concepts used to describe how human experts perform the task (problem solving) domain-independently. Task ontology could be what we need to make knowledge-based systems aware of what they know about the task they are performing, and of what conceptualization the knowledge in the knowledge base is based upon, etc. Research on ontology from an engineering point of view is called ontological engineering.

Thus, knowledge engineering has developed into ontological engineering (Mizoguchi, Sinitsa and Ikeda, 1996a; Mizoguchi, 1998). It is true that knowledge is domain-dependent, and hence knowledge engineering, which directly investigates such knowledge, has been suffering from a rather serious difficulty caused by its specificity and diversity. However, ontology research is different. Ontological engineering investigates knowledge in terms of its origins and elements from which knowledge is constructed. The hierarchical nature of concepts and the decomposability of knowledge are exploited to deeply investigate primitives of knowledge as well as background theories of knowledge which enables us to avoid the difficulties that knowledge engineering has been faced with.

Here, by a task, we mean a problem solving process like diagnosis, monitoring, scheduling, design, and so on. In our context, instruction is a task, as is supporting the learning process. A task ontology is obtained by analyzing the task structures of real world problem solving. It does not cover the control structure but do components or primitives of unit activities taking place during the performance of the tasks. The ultimate goal of task ontology research includes providing a theory of all the concepts necessary for building a model of the human problem solving processes.

3.2 Computational semantics of an ontology

What are the computational semantics of an ontology? Are they just a set of terms? This is one of the most crucial points of an ontology. Regardless of the fact that an ontology sometimes looks like just a set of terms, it has richer computational semantics. One of the authors has proposed the following three levels of ontologies (Mizoguchi, 1998).

- Level 1: A structured collection of terms. The most fundamental task in ontology development is articulation of the world of interest, that is, elicitation of concepts and identifying the so-called is-a hierarchy among them. These are indispensable things to an ontology. Typical examples of ontologies at this level include topic hierarchies found in internet search engines and tags used for metadata description. Little definition of the concepts is made.

- Level 2: In addition to the contents of a level 1 ontology, we can add formal definitions to prevent unexpected interpretation of the concepts and necessary relations and constraints also formally defined as a set of axioms. Relations are much richer than those at the level 1. Definitions are declarative and formal to enable them to be interpretable by computers. The interpretability of an ontology at this level enables computers to answer questions about the models built based on the ontology. Many of the ontology building efforts aim to build ontologies at this level.

- Level 3: The ontology at this level is executable in the sense that models built based on the ontology run using modules provided by some of the abstract codes associated with concepts in the ontology. Thus, it can answer questions about the runtime performance of the models. Typical examples of this type are found in task ontologies. Software components in component ware roughly correspond to an ontology at this level. But, they have nothing corresponding to levels 1 and 2.

3.3 Knowledge engineering of authoring

Roughly speaking, authoring consists of "static knowledge" organization and "dynamic knowledge" organization. The former includes curriculum organization with instructional design and the latter tutoring strategy organization for adaptation to the learners. Many of the ITS-related research published thus far is concerned with the latter because the intelligent behavior of ITSs emerges not out of the curriculum itself but out of adaptive tutoring strategies based on the learner model constructed. This is one of the major reasons why ITS researchers, who have strong AI bias, do not pay much attention to instructional design technology and have employed heuristics-based approaches to the knowledge organization.

On the other hand, curriculum authoring should be supported by instructional design based on instructional science which relies on learning science. This dependency on several types of knowledge is the source of intelligence of authoring systems which are expected to help authors build curriculum systematically and scientifically. Declarative representation of such a system of knowledge enables really intelligent authoring systems to be designed.

Note that not all of the knowledge necessary for an IIS can be theory-based. Just like a knowledge-based system, a good combination of heuristics and domain theories is necessary in practice. However, the more advancements made in ID and IS research, the more the heuristics can be explained and hence supported by theories. ID ontology-aware environments could help find justifications of heuristics. This topic is discussed again in section 5.

3.4 How an ontology provides us with a solution

First of all, a level 1 ontology provides a set of terms which should be shared among people in the community, and hence could be used as well-structured shared vocabulary. These terms enables us to share the specifications of components' functionalities, tutoring strategies and so on and to properly compare different systems.

An ontology is also defined as "an explicit specification of a conceptualization" (Gruber) which suggests that it explicitly represents the underlying conceptualization which has been kept implicit in many cases. A level 2 ontology is composed of a set of terms and relationships with formal definitions in terms of axioms. Such axioms are declarative, and hence such an ontology represents the conceptualization declaratively. Thus, a level 2 ontology is the source of intelligence of an ontology-based system.

Another role of an ontology is to act as a meta-model. A model is usually built in the computer as an abstraction of the real target. An ontology provides concepts and relationships which are used as the building blocks of the model. Axioms give semantic constraints among concepts. Thus, an ontology specifies the models to build by giving guidelines and constraints which should be satisfied. This is how the function is viewed at the meta model level. Needless to say, this characteristic is what an authoring system really needs. In fact, we can find some research based on the meta-model function of an ontology (Mizoguchi, Sinitsa and Ikeda, 1996b; Ikeda, Seta and Mizoguchi, 1997; Murray, 1998; Chen, Hayashi, Kin, Ikeda, and Mizoguchi, 1998).

A shared ontology is a first step towards standardization. Not only informal definitions of terms/concepts but also intermediate concepts are made explicit by the level 1 ontology. The structuring usually employs is-a and part-of links to relate concepts to each other. The structure obtained in a level 1 ontology itself represents an understanding about the domain of the developer. It is usually much more informative than definition of a term. An ontology cannot instantly become a standard. An ontology designed gives a test-bed for establishing a standard.

On the basis of standardized terms and concepts, knowledge of the domain can be systematized in terms of the concepts and standardized relationships identified in the ontology. This is what we are intending to do in our ambitious plan “building an ontology of Instructional Design” which makes an authoring system “ID-theory-aware”.

4. ONTOLOGICAL ENGINEERING OF ID

While the previous sections were devoted mainly to ontological engineering as an enabling technology, this section is concerned with ID/IS knowledge, which is the target knowledge of ontological engineering.

4.1 Necessity for an ontology of instruction

Although many AI-ED researchers are well aware of instructional theories, they encounter difficulties in implementing systems that can rely on a unified and complete set of concepts and principles. How can the gap between the two be bridged? An IIS needs terms/concepts concerning pedagogical actions to ground the functionality onto concrete actions. The justification of an IIS should be given by theories and the source of intelligence of the systems should come from the knowledge bases containing this knowledge. However, idiosyncratic implementations dominate real system development, and this may be a major source of difficulty in the lack of interoperability between knowledge bases and systems. Dissemination of Instructional Knowledge and the sharing of this knowledge between humans and computers has become a necessity. Easy access to instructional theories is of worth to both human and computer agents: for humans, conventional browsers are enough; for computers, somewhat deeper operationality is required; Ontological Engineering helps specify the higher level functionality of IISs; it allows us to bridge the gap between human knowledge and the knowledge in the knowledge bases. The authors believe that a new direction is needed for AI-ED research in the XXIst century, which could take the form of an Ontology of Instruction, that could pave the way to the building of an ID-aware Authoring Environment for IISs. An authoring agent could explain relevant theories in response to the author’s request; it could give the author some possible justifications for teaching and learning strategies from a theoretical point of view. To this end, we envision the building of an Instructional knowledge server on the Web, having such support functionality, called an “Instructional Ontology-aware environment”. In order to reach this goal, we have determined that we first need to extract an ontology from existing Instructional theories and from Instructional Design models.

4.2 What is Instructional Science and what is Instructional Design?

Instructional Science was born in 1966, with Bruner’s Theory of Instruction, and has grown remarkably since then with work from Ausubel, Glaser, Gagné, Merrill, to name but a few, and with recent developments in Cognitive Sciences (see ID bibliography, http://www.unc.edu/cit/guides/irg-22.html). It builds upon Learning Sciences (Psychology of Learning, Sociology of Learning, Systems Science), and consists of theories, models and methodologies for Instruction and for Research on Instruction. As defined by Simon, Instructional Science is a Design Science. It has both descriptive and prescriptive components; the prescriptive part forms what is called Instructional Design. Instructional Science contains and builds theories, models and methodologies. The focus of studies in Instructional Science is the interrelationship between four classes of variables: instructional situation, subject-matter, instructional outcomes, and instructional strategy variables (Reigeluth, Buderson and Merrill, 1994).

Instructional Design is the systemic and systematic process of applying strategies and techniques derived from behavioral, cognitive, and constructivist theories to the solution of instructional problem. It represents the systematic application of theory and other organized knowledge to the task of instructional design and development. Subprocesses include: analysis (of subject-matter, goals and objectives, student characteristics, context and constraints), design (higher level decision-making based on strategies), development (lower level decisions based on learning and assessment activities and material), evaluation (of process and product), delivery (product and services) and management (of design and of delivery processes). Instructional Design is domain-independent, generic and theory-based; it contains concepts, rules or principles. The state of the art in Instructional Design shows concerns about unification and integration (Seels, 1995; Duchastel, 1998), as well about taxonomic issues (Seels, 1997), toward a better integration of taxonomic concepts between learning domains - affective, cognitive and psychomotor. Instructional Design is independent from learning paradigms (Lebow, 1995), and evolves along with new instructional paradigms. Reigeluth's claim for a new paradigm of Instructional Theory (Reigeluth, 1996) contains the following keywords: customization, autonomy, cooperation, shared decision-making, initiative, diversity, networking, holism, process-oriented, and Learner as "King"!

4.3 From teaching and tutoring strategies to instructional strategies

Can teaching strategies be imported from their natural setting, the classroom milieu, to IISs? Teaching strategies have been studied as strategies employed by teachers in a situation with the following characteristics: the actors are one teacher and a group of students; all meet regularly in the same place at the same time, on a yearly basis; their encounters are determined by a school system and by established relationships (authority); personal development, socialization and instruction are important goals, sometimes prior to instruction; there is a culture of teaching, with a set of expectations, which vary with the culture of a society or milieu, etc. Teaching strategies can therefore be called "knowledge in context", and are strongly dependent on context. Teaching expertise is defined as problem-solving expertise in the context of a classroom, with two main components: routine knowledge and creative knowledge, both strongly related to the classroom milieu (Gagné, Yekovich and Yekovich, 1994). As a conclusion, Teaching strategies should be reconsidered under the light of "knowledge in context" before being imported into an IIS; moreover, they may be analyzed and selected only on the basis of their relevance for IISs and of congruence; some may be strongly revised or adapted and others eliminated completely.

Tutoring refers to the situation of one tutor interacting with either one or a few students, and etymologically contains the notion of strong guidance toward an explicit goal; the metaphor of the tutor for a plant illustrates the idea of going in a straight direction. Tutoring has proved to be the most effective strategy to ensure learning, as demonstrated by Bloom in his Mastery Learning research, and his challenge to discover strategies as effective as tutoring, for group learning still remains unanswered. Tutoring therefore provides a solid ground for ITS research. When referred to in ITS research, tutoring has more flexible semantics, representing actions to be taken by a "knowledgeable" system to support or guide learning by an individual student. However, it remains squarely in the field of individual learning, as does traditional distance learning.

Individual learning, individual tutoring and asynchronous communication are typical features of a distance learning situation, with two main implications for distance learning systems: extensive macro- and micro-instructional design, and a strong student support system. Distance Learning and Instructional Design are intrinsically related (Bourdeau and Bates, 1996), and knowledge gained in distance learning theory and design (Moore and Kearsley, 1996) provides direction for the design of OLEs. Hybrid systems are what we need to envision, where all actors, students and teachers, share both "live" events in natural settings, and virtual ones, with a system that is aware of both universes, and capable of referring to both in its reasoning and decision making.

The term telelearning is used to designate new forms of distance or of computer mediated learning, where the distance is not only distance in space or time as in traditional distance learning, but the mediation in learning activities is served by media such as multimedia shared workspaces, multimedia communication, and multimedia servers. Many variations can be found in terms of presence, telepresence, meeting in virtual spaces, interactivity with rich multimedia environments, and extensive human interactions in a virtual world with no limits except access and language. Collaborative telelearning emphasizes the collaborative interaction between students in a virtual world (Bourdeau and Wasson, 1997). Instructional Design of Simulations and Labs for Telelearning means interdependent design of learning scenarios and learning environments (Bourdeau, 1998).

There is more to Instructional Strategies than individual tutoring, and a rich set of appropriate strategies could be known by an intelligent authoring environment, along with the conditions for their appropriate use,; these include new ideas such as Reigeluth's learner-focused instruction with appropriate combinations of challenge and guidance, empowerment and support, self-direction and structure, where learners may choose from methods such as problem or project-based learning, simulations, tutorials, and team-based learning (Reigeluth, 1996).

4.4 Instructional design-aware authoring

Instructional Knowledge has been used in the field of ITS for approximately a quarter of a century, and experience gained in the building of ITSs shows that they are often curriculum or topic oriented, sometimes have weak congruence between analysis of task and instructional or assessment strategies, lack awareness of the learner context besides cognitive aspects; learner modelling is oriented toward control. Existing instructional knowledge is fragmental and sometimes used more to serve technical design needs rather than learning needs. Recent efforts in the AIED community appear to be directed toward ITS-Authoring (Nkambou, Gauthier, and Frasson, 1996; Murray, 1996; Redfield, 1997; Ikeda, Seta and Mizoguchi, 1997; Chen, Hayashi, Kin, Ikeda, and Mizoguchi, 1998; Devedzic, V. Jerinic, L. and Radovic, D., 1999). Murray's recent review (Murray, 1998) shows trends toward the inclusion, if not integration, of four components: Tools for Content, Instructional Strategy, Student Model, and Interface Design.

Given the characteristics of Instructional Design (domain-independent, theory-based), efforts have been made in the last decade toward the modelling of this knowledge (Tennyson and Barron, 1995). A major effort was conducted at USAF Armstrong Lab in the 80s as leaders in the field gathered over several years with the goal of building a consensual body of Instructional Design knowledge, and of automating the Instructional Design Process. This project, called AIDA increased the awareness of the necessity of consensual knowledge; other major projects include ID Expert by Merrill, GUIDE by Tennyson, System Dynamics at U. Bergen, SAFARI at U. Montréal and AGD at LICEF (Paquette and Girard, 1996). Instructional Design knowledge also served as a foundation for building the object-oriented Virtual Campus Environment at LICEF (Paquette, Ricciardi-Rigault, Bourdeau, Paquin and Liégeois, 1995).

Intelligent Authoring Environments that can support the building of ITSs need a solid foundation in Instructional Science, with a coherent set of concepts and principles for building quality products. Such environments should provide authors with a choice of either long established knowledge or more recent developments, such as Reigeluth's proposal to consider learners as co-designers of their instruction, where "learners have the capability to request that the computer system use particular instructional strategies, as well as the computer deciding which strategies to use based on learner input" (Reigeluth, 1996).

4.5 Designing and authoring an IIS, ILE, or OLE

An ID-aware Authoring System knows the distinction between designing an IIS, an ILE, and an OLE; it provides the set of requirements and decisions to be made in each case before starting any authoring, in order to create a complete, coherent and congruent product.

The requirements for designing an IIS rely on the knowledge of individual dynamic characteristics as much as of the didactic knowledge. We expect ITS research to evolve towards multimedia and VR technology, e.g. for simulations and labs; it could therefore benefit from being based on explicit and theory-based design principles, e.g. designing learning scenarios together with learning environments. Explicit statements would be used to specify the conditions of learning for which the system has been designed complementary to, supplementary to or replacement for teaching.

Designing an ILE requires a different set of decisions that can refer to either individual or team-based learning with a philosophy such as situated learning. Having fundamentals for a constructivist design, for example, helps us in making explicit statements about the design principles used, the authoring decisions made, and about their pedagogical finality and effectiveness (Lebow, 1995).

Designing an OLE contains challenges that seem to be particularly in phase with the time spirit as we step into the XXIst century. Being “open” can mean keeping your eyes open, and also being open-minded. What does it mean for an OLE? Requirements for an OLE typically are: 1) to know about external learning events, both those planned and the ones that really happen, 2) to be able to reason, make hypothesis and decisions based on both internal and external events and 3) to be flexible in adapting instructional strategies based on culture or affects.

Instructional Design has the capability to make ITSs and ILEs evolve toward Open Learning Environments as follows: Instructional Design is a process by which learning events can be defined or described, independently of their instructivist or constructivist orientation. In order to prevent the « box effect » of Intelligent Tutoring – i.e. a student interacting « within » a system that has no knowledge of any external event- Instructional Design provides a way to inform the system about external learning events, so that it can take these events into account in its reasoning and decision-making. This bridging could make ITSs more « real » to students, producing a better learning companion or tutor.

5. A ROAD MAP

The following is a road map towards the new direction. Challenges are threefold: 1) the sharing among humans, and through computer technology, of the knowledge we have accumulated thus far. 2) The sharing extended from among humans to among computers, 3) The operationalization of this knowledge to support the building of IISs.

The first challenge is to have computers mediate the sharing of our knowledge. In order to enable computers to mediate humans in knowledge sharing, there has to be more than an information retriever. If the knowledge is derived from different conceptualizations of the world of interest, and unrelated terms are used to represent the knowledge, computers cannot do the mediation. They need at least to share the fundamental conceptualization of the target world with humans, and a common vocabulary for representing the knowledge, in order to do meaningful mediation using the common terms. The level 1 ontology discussed in 2.1 plays a sufficient role in this goal. This would be the first step in our enterprise. It is important to note that “shared vocabulary” does not mean excluding the variety of theories based on different viewpoints, nor does it mean standardization of knowledge. The goal is to share primitive concepts in terms that we can use to describe the knowledge and theories, in full respect of their integrity. We anticipate that most of the Instructional knowledge can be described in terms of a shared vocabulary. We can see movements related to this enterprise: On-line glossary of ID (e.g., http://garnet.acns.fsu.edu/~www6982/glossary.html) and home page of ID theories (e.g., http://www.cudenver.edu/~mryder/itc_data/theory.html); research on taxonomy of ID knowledge (Seels, 1997) also gives a positive indication. One further step towards the level 1 ontology would make a concrete contribution to the goal.

The second challenge is to extend this sharing from among humans to among computers. Since a shared vocabulary is not rich in meaning - it consists only of a set of common terms structured in terms of is-a and part-of links resulting in a semantically poor situation, computers cannot go deeper. To enhance the computer’s intelligence, the level 2 ontology introduces definitions of each term and richer relationships than in level 1, using axioms. An axiom relates two concepts semantically, which allows computers to partially understand the rationale of the configuration of the world of interest, in this case learning and instruction.

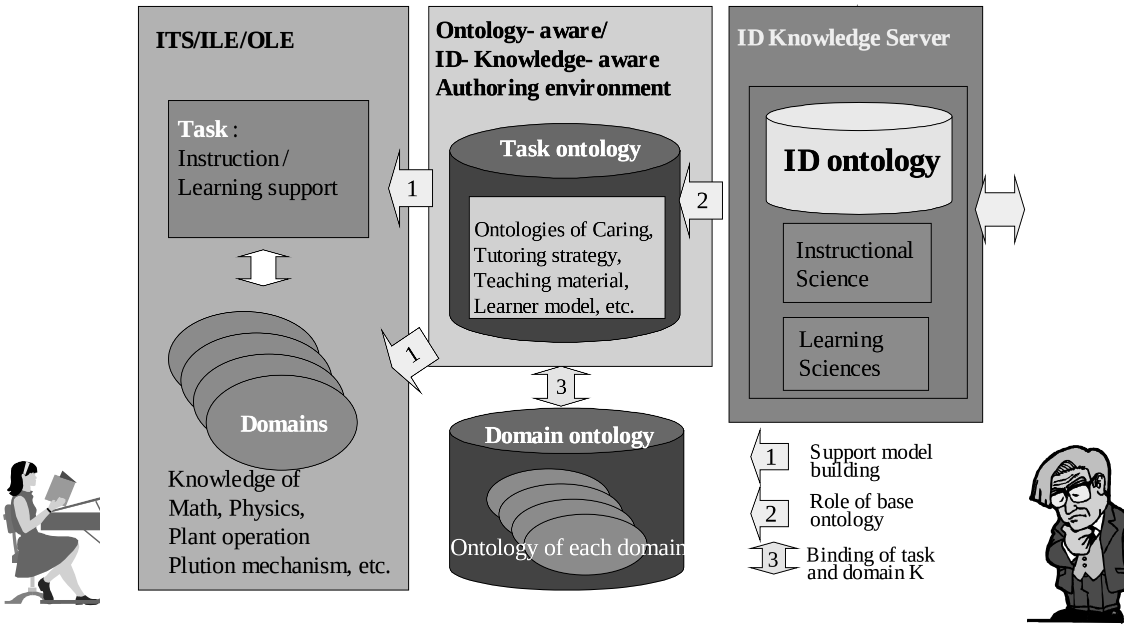

Figure 1. An ontological diagram of related systems with ID knowledge servers.

The operationalization of this knowledge leads to the building of IISs. This requires a level 3 ontology to enable computers to run the code corresponding to the activity-related concepts. Knowledge at this level is mainly concerned with task ontology which contains concepts of action of the system in performing a specific task (instruction, learning support). An ontological diagram of IIS, authoring environment and an Instructional Design knowledge server is shown in Fig. 1. When we have a shared vocabulary, the ID-knowledge server communicates with humans who need help in finding ID knowledge appropriate for their goals; it can give justifications to an ontology in ID-knowledge-aware authoring environments. Thus, such authoring environments can discuss the appropriateness of strategies adopted with authors helped by the ID knowledge server. Future IISs developed in this way would assist the seamless flow of knowledge from the designers to the learners.

One step in this direction could be to build upon an existing courseware engineering workbench such as AGD (Paquette, 1996), to isolate one set of ID decisions and re-engineer them based on task ontology; a micro-domain ontology could be built from meta-models related to this set. This preliminary work could illustrate and test the idea of an ID Ontology server.

6. OTHER PROMISING DIRECTIONS

6.1 Design patterns (Gamma, Helm, Johnson and Vissides, 1994; Devedzic, 1999; Shimizu, Seta, Hayashi, Motomatsu, Ikeda and Mizoguchi, 1999)

Axioms in an ontology give semantic constraints among concepts along with rigorous definition of concepts. They play the role of constraints which the models built must satisfy. An ontology-aware authoring system can thus generate warning messages to the authors when a constraint is violated. The constraints could work as guidelines to the model-building process. This is the source of the intelligence of such an authoring system. However, we have to pay an attention to the fact that not all of the guidelines have to be satisfied by all of the model-building processes. As described in 3.3, authoring still requires heuristics. That is, there can be soft heuristic guidelines which are sometimes better to satisfy but not an essential, supported not by theory but by experience and so forth. These kinds of suggestions are not appropriate for representation in the form of axioms. Axiomatic representation is appropriate only for theories.

In the object-oriented paradigm, design patterns are becoming popular. To avoid confusion between instructional “design” and “design” pattern, in this paper, we call the latter “heuristic design patterns”. Heuristic design patterns are patterns of class configurations which represent a chunk of information useful for design decisions in configuring classes and objects. They are obtained by abstracting the rich experiences of object-oriented modeling and can provide suggestions useful during the course of model building. We can use the heuristic design patterns of IIS modeling for the authoring systems on top of the ontology-guided and ID-theory-based model construction to generate helpful suggestions.

The use of heuristic design patterns in our plan is different from that in the Object-oriented paradigm, given the declarative nature of an ontology. Heuristic design patterns in ontology-aware authoring systems are represented in terms of an ontology, and hence they are operational, while those in the object-oriented paradigm are not operational because they are informal assuming human interpretation. All of the terms appearing in an level 2 ontology are operational in the sense that computers can interpret the axioms associated with the terms used in the design patterns so that the applicability of the patterns to the current situation can be checked by computers with appropriate suggestions. Application of design patterns to IISs were recently discussed by V. Devedzic (Devedzic, 1999) and H.Shimizu, et al.(Shimizu, Seta, Hayashi, Motomatsu, Ikeda and Mizoguchi, 1999).

An ID knowledge server and a heuristic design pattern server would realize an effective combination of theories and heuristics for intelligent authoring environments.

6.2 XML-based documentation of IS knowledge

XML (XML) is a simplified version of SGML which is a powerful markup language definition language and has been widely used in document description for many years. XML is equipped with several powerful hyper-reference functions to fit the internet environment. One of the good things of XML includes is that the users can define their own tags to indicate not only structural information about the document but also the semantic information for various uses of the document to enable semantic interoperation. A set of XML tags are a kind of level 1 ontology.

XML is sometimes said to be HTML++ because XML is so flexible that it enables us to present XML documents in any way that we like, while HTML documents are used only for the presentation we see in Web browsers due to its predefined tags. A first step of use of the XML tags might be to design metadata for Learning objects which has been done in the IEEE LTSC committee P1484 (IEEE LTSC), IMS (IMS) and ARIADNE (ARIADNE). A more advanced application of XML tags might include designing XML tags for making documents be usable as live teaching material.

For example, we can design an intelligent instructional player for an XML document defined as a teaching material by sharing a set of tags for explicating the instructional roles of the portions of the document and controlling the interpretation. It could “Play” the XML document while adapting to the performance of the learners with the help of Java applet. Further, we could formalize the set of tags to specify the performance of such instructional players. Intelligent players with plug & play capability and shared XML tags are expected to be promising alongside a suitable Java implementation.

The Resource Description Framework: RDF (RDF) enables richer and more solid marking than XML. XML with RDF will open a new world of instructional documentation which can produce both documents for human interpretation and those for computer interpretation to make them teaching materials. The critical issue here is to design a set of sophisticated tags which we can share as a standard. This requires a huge effort but it should be rewarding. Ontological engineering can make a substantial contribution to this enterprise.

7. CONCLUDING REMARKS

We have discussed ontology-awareness in AI-ED research. The discussion was started by an analysis of the current status of AI-ED research in terms of intelligence, conceptualization, standardization and theory-awareness which suggest an importance of ontology-based architecture with appropriate ontologies. Ontological engineering of IS/ID was then discussed, followed by a road map towards an ontology-aware authoring system. Heuristic design patterns and XML-based documentation were also discussed.

The implications of “ontology-awareness” are deep, not only for AI-ED research into system building but also for knowledge sharing between humans and computers. Most of the existing theories are in the form of informal representation requiring human interpretation. One of the most important features introduced by ontological engineering is that is enables humans to share the theories with computers. Furthermore, it can mediate people to find a minimum agreement.

We hear some voices saying “It’s OK that an ontology represents a shared conceptualization in a community. But, let me tell you that ID is the most dangerous domain to design an ontology because it is almost impossible to find a consensus.”

Our answer is: Yes, but, that is the very reason why we advocate ontological engineering of ID. Our claims include an ontology helps people identify what they agree on and what they do not. An ontology is not the total knowledge of the target world but is a backbone/skeleton of the target world. Each person could come up with their own ontology developed by an ontology editor with sophisticated guidelines. Imagine that an "Ontology-mediated agreement-finding system" was available. Then, people would start discussions to find similar and different positions, with the help of the "Ontology-mediated agreement-finding system". They can improve discussion by exploiting the hierarchical nature of the ontologies which help them find easy agreeable general concepts by going up the hierarchy. The is-a and part-of hierarchies of the ontology will help people involved in agreement finding notice what are the essential differences between them. Each ontology can be interactively reorganized according to the goals or purposes specified. They might notice what should be stable concepts to make the ID research meaningful as well as what could be left unstable and dependent on personal views of the world. The important thing is that what they finally get is an ontology which represents the minimum amount of agreement at the time. It may be far from complete, but it should provide a good start to come up with richer agreement.

References

ARIADNE Project, http://ariadne.unil.ch/

Berliner, D. (1989) Furthering Our Understanding of Motivation and Environments, in Ames, C. & Ames, C. (Eds.), *Research on Motivation in Education*, vol. 3, Goals and Cognitions. Academic Press.

Bourdeau, J. and Bates, A. (1996) Instructional Design for Distance Learning, in *Journal of Science Education and Technology*, 5, 4, 267-283.

Bourdeau, J. and Wasson, B. (1997) Orchestrating collaboration in collaborative telelearning. In B. du Boulay & R. Mizoguchi (Eds.) *Proc. of the 8th World Conf. on AI-ED*, 565-567. Amsterdam: IOS Press.

Bourdeau, J. (1998) Instructional Design of VR Simulations and Labs, *Tele-learning’98*, Vancouver, Nov.15th.

Breuker, J. and de Velde, V. (1994) *The Common KADS Library for Expertise Modelling*, IOS Press, Amsterdam.

Chandrasekaran, B. (1986) Generic tasks for knowledge-based reasoning: the right level of abstraction for knowledge acquisition, *IEEE Expert*, 1, 23-30.

Chandrasekaran, B. Josephson J. R. and Benjamins, R. (1999) What are ontologies, and why do we need them?, *IEEE Intelligent Systems*, Vol.14, No.1, 20-26

Chen, W. Hayashi, Y. Kin, L. Ikeda, M. and Mizoguchi, R. (1998) Ontological Issues in an Intelligent Authoring Tool, in Chan T-W., Collins A. & Lin J. (Eds.), Global Education On the Net, *Proc. of ICCE'98*, vol. 1, 41-50.

Clancey, W. J. (1985) Heuristic classification, *Artificial Intelligence*, 27, 3, 289-350.

Devedzic, V. Jerinic, L. and Radovic, D. (1999) The GET-BITS Model of Intelligent Tutoring Systems, *International Journal of Continuing Engineering Education and Life-Long Learning* (to appear).

Devedzic, V. (1999) Using design patterns in ITS development, Proceedings of *AIED99*, Le Mans, France, 657-659.

Duchastel, P. (1998) Prolegomena to a theory of instructional design. *ITFORUM*, September 1998, online. http://itech1.coe.uga.edu/itforum/paper27/paper27.html .

Gagné, E., Yekovich, C. and Yekovich, F. (1994) *The cognitive Psychology of School Learning*. Harper Collins College Publ., 2d ed.

Gamma, E., R. Helm, R. Johnson and J. Vissides, (1994) *Design patterns*, Addison-Wesley Publishing Company

Genesereth and Nilsson. (1987) *Foundation of Artificial Intelligence*. Morgan Kaufmann Publishers, Inc.

Gruber, T., What-is-an-ontology? http://www-ksl.stanford.edu/kst/what-is-an-ontology.html.

IEEE LTSC, Learning Technology Standards Committee, http://grouper.ieee.org/ltsc/

Ikeda, M. Seta, K and Mizoguchi, R. (1997) Task Ontology Makes It Easier To Use Authoring Tools. *Proc. of IJCAI-97*, Nagoya, Japan, 342-347.

IMS, http://www.imsproject.org/

Keller, C. (1983) Motivational Design of Instruction, in Reigeluth, C. *Instructional Design Theories and Models: An Overview of their current status*. Mahwah, NJ: Lawrence Erlbaum Associates.

Lebow, D. (1995) Constructivist Values for Instructional Systems Design: Five Principles Toward a New Mindset, in Seels, B.. *Instructional Design Fundamentals: a Reconsideration*. Engl.Cliffs, NJ: Ed.Tech. Publ.

Mizoguchi, R. Tijerino, Y. and Ikeda, M. (1992) Task ontology and its use in a task analysis interview system -- Two-level mediating representation in MULTIS --, *Proc. of the JKAW'92*, pp.185-198.

Mizoguchi R. Vanwelkenhuysen, J. and Ikeda, M. (1995) Task Ontology for Reuse of Problem Solving Knowledge. *Building & Knowledge Sharing 1995 - KB&KS'95, 2nd International Conference on Very Large-Scale Knowledge Bases*, Enschede, The Netherlands, pp.46-59.

Mizoguchi, R. and Ikeda, M. (1996) Ontology Engineering-Towards the Basic Theory and Technology for Content-Oriented Research-, *J. Jpn. Soc. for Artificial intelligence*, Vol. 12, No. 4, pp. 559-569, http://www.ei.sanken.osaka-u.ac.jp/pub/miz/miz-onteng.pdf.

Mizoguchi, R. Sinitsa, K. and Ikeda, M. (1996a) Task ontology design for intelligent educational/training systems, *Workshop on Architectures and methods for Designing Cost-effective and Reusable ITSs, ITS96*, Montreal. http://advlearn.lrdc.pitt.edu/its-arch/papers/mizoguchi.html.

Mizoguchi, R. Sinitsa, K. and Ikeda, M. (1996b) Knowledge engineering of educational systems for authoring systems design, *Proc. Euro AI-ED*, pp.329-335.

Mizoguchi, R. Ikeda, M. and Sinitsa, K. (1997) Roles of Shared Ontology in AI-ED Research -- Intelligence, Conceptualization, Standardization, and Reusability. *Proc. of AIED-97*, Kobe, Japan, 537-544.

Mizoguchi, R. (1998) A Step towards Ontological Engineering. http://www.ei.sanken.osaka-u.ac.jp/english/ step-onteng.html

Moore, M. and Kearsley, G. (1996) *Distance Education: A Systems View*. Wadsworth Publ.

Murray, T. (1996) Special Purpose Ontologies and the Representation of Pedagogical Knowledge., *ICLS 96*.

Murray, T. (1998) Authoring Knowledge-Based Tutors: Tools for Content, Instructional Strategy, Student Model, and Interface Design, in *Journal of Learning Sciences*, 7 (1), 5-64.

Nkambou, R. Gauthier, R., & Frasson, C. (1996) CREAM-tools: An authoring environment for curriculum and course building in an ITS. *Proc. of the 3d Int'l Conf. on CAL & Inst'l Sc. & Eng*, NY: Springer-Verlag.

Paquette, G, Ricciardi-Rigault, C., Bourdeau, J., Paquin, C. and Liégeois, S. (1995) Modelling the Virtual Campus for Interactive Distance Learning, *Proc. of EDMEDIA'95*. Charlotteville, VA: AACE, 523-528.

Paquette G. and Girard J. (1996) AGD: a course engineering support system, *Proc. of ITS-96*, Lecture Notes in Computer Science, 1086:382-391. Springer-Verlag

Puerta, A.R and Musen, M.. (1992) A multiple-method knowledge-acquisition shell for the automatic generation of knowledge-acquisition tools, *Knowledge Acquisition*, 4, 171-196.

RDF, http://www.w3.org/RDF/FAQ

Redfield, C. (1997) An ITS authoring tool: Experimental advanced instructional advisor. In *Papers from the 1997 AAAI fall symposium* (Tech. Rep. FS-97-01, 72-78). Menlo Park, CA: AAAI.

Reigeluth, C. Buderson, C. and Merrill, D. (1994) Is there a Design Science of Instruction? in Merrill, M. D. and Twitchell D.G., ed. *Instructional Design Theory*. Englewood Cliffs, NJ: Ed. Tech. Publ.

Reigeluth, C. (1996) What Is the New Paradigm of Instructional Theory. *ITFORUM*, online. http://itech1.coe.uga.edu/itforum/paper17/paper17.html .

Seels, B. (1995) *Instructional Design Fundamentals: a Reconsideration*. Engl.Cliffs, NJ: Ed.TEch. Publ.

Seels, B. (1997) Taxonomic Issues and the Development of Theory in Instructional Technology. *Educational Technology*, Feb., 12-21.

Self, J. (1997) Communication at the Conference of AIED-97, Kobe, Japan.

Seta, K. Ikeda, M. Kakusho, O. and Mizoguchi, R. (1997) Capturing a Conceptual Model for End-user Programming -Task Ontology as a Static User Model- , *Proc. of The Sixth International Conf. on User Modeling '97* (UM97), 203-214.

Shimizu, H., K.Seta, Y. Hayashi, S. Motomatsu, M. Ikeda and R. Mizoguchi (1999) A basic consideration on design patterns for ontology building, *Proc. of the 58th National Conference of Information Processing Soc*. Of Japan, 3U-9.

Tennyson, R. and Barron, A., Eds. (1995) *Automating Instructional Design: Computer-Based Development and Delivery Tools*. Springer, NATO ASI Series, Series F: Computer and Systems Sciences, vol. 140.

XML, eXtensible Markup Language, http://www.w3.org/XML/