IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITP5221

This paper appears in *Managing Modern Organizations Through Information Technology*, Proceedings of the 2005 Information Resources Management Association International Conference, edited by Mehdi Khosrow-Pour. Copyright 2005, Idea Group Inc.

Towards an e-Learning Model of Self-Tutoring

Rebecca Grasser and Santosh K. Misra
Cleveland State University, Cleveland, OH 44114, USA, {r.grasser, s.misra@csuohio.edu}

INTRODUCTION

There are many definitions of e-learning (McArthur, 2003). In general, e-learning includes any electronic application used for instruction. This can range from Web-displayed: a static site for displaying readings and course administration to fully Online: a site where there is no face-to-face interaction between participants. In all these cases, computer technologies have been deployed to facilitate and speed the delivery of knowledge. Increasing acceptance of computer-enhanced delivery of knowledge has spawned alternative techniques for providing education. Computer-enhanced delivery techniques now include virtual learning environments, interactive computer-based curriculum as well as online delivery methods where the teacher and the student may never meet. E-learning can, therefore, allow a wide audience to receive training in a cost-effective manner (DeRouin, 2004).

A number of e-learning models have been proposed in the literature. For example, MacDonald et al. (2001) propose a demand driven framework for web-based learning. This framework incorporates both learner variables as well as institutional variables. Learner variables include learner needs, motivation, environment, evaluation and convenience; institutional variables include program goals, program evaluation strategies as well as strategies for continuous improvement.

Deubel (2003) notes that a learning model should be chosen before the system is designed. Instruction design learning models include the apprenticeship, incidental, inductive, deductive, and discovery models. The apprenticeship model presents the material in a procedural manner. This can be contrasted to the incidental model which describes an event or scenario to develop the concepts. The inductive approach starts small (the concepts) and moves to the large, while the deductive model presents the information in the opposite manner. Finally, the discovery model can be compared to problem-based learning or other forms of inquiry learning.

At the Open University in the UK, Mason (1998) proposes a framework containing three interrelated models. These include the content + support model, a system that separates the course content from the tutorial/collaborative portion of the course. The wrap around model allows the content developer to create new material that wraps around existing texts and content modules. The focus is shifted towards the learner, with collaboration taking approximately one-half of the course time. In the integrated model, the course is nearly exclusively learner centered, with significant learner collaboration.

Another set of models focus on the instructional strategies that emphasize the need for careful consideration of the underlying pedagogy and the mechanics of online learning (Engelbrecht, 2003). As argued by Conrad (2000), effective e-learning can occur when instructional practices and Internet capabilities are integrated to direct a learner toward a specified level of proficiency in a specific competency. Instructional value can also be added by customizing content for the needs of the learners. (Engelbrecht, 2003)

Online learning is defined by some as a process that takes place at the intersection of social and technological systems (Seufert et al., 2002). Social and technological system proponents do not consider learning as a passive activity but that of interaction through the use of online communication tools. Online communication tools facilitate management of knowledge since up-to-date information can be quickly incorporated into the environment with relative ease. Online tools can help in the learning method of the learner. It allows the learner access to quantity of information anytime anywhere without the risk of self-esteem problems of failure to learn. The learner gets objective evaluation of his/her performance and is allowed the time necessary to master the topic at hand.

We should note that technology alone cannot substitute content and the context of learning. Anytime anywhere access and privacy of learning sessions afford a degree of comfort to the learner facilitating the learning process. What then should be the goals of such a learning system? We propose the following goals for such a system:

- Must present factual knowledge regarding the domain of study and support creative problem solving by the learner;
- Support a subject-oriented learning.
- Support self-guiding capabilities to control the speed and the order of the knowledge acquisition process.

Seufert et al. (2002) have three categories of studying (or learning). These are contact studies, self-studies, and context-studies. Contact studies are areas where face-to-face contact and instructor-centered teaching (i.e. lectures, question-and-answer sessions) are the primary means of learning. Self-studies involve less teaching and more guiding. The learner works independently to solve problems. Finally, context-studies involve the learners own creation of knowledge. Learners develop critical reflection skills while attempting real-world, multidisciplinary problems. The model we are proposing is designed for self-studies.

In this paper, we present a self-learning model that seeks to incorporate the above principles. In section 2, we describe an expert domain model that helps a learner in complex learning environment. We apply this model in section 3 to a Database Normalization problem that often proves challenging to undergraduate and graduate students. Section 4 describes a proposed approach to validate the model. We conclude the paper by describing future plans to implement and test the model in an experimental setting.

EXPERT DOMAIN E-LEARNING MODEL

We propose an e-learning model as shown in Figure 1. The input for the model is a set of facts and rules provided by the instructor.

The domain model is a usable representation of the information to be taught or reinforced. The system needs to be able to parse the domain model information so that material can be easily presented, feedback generated, and problems reinforcing the material can be created and selected. The output of the domain model can include concepts, facts, heuristics, examples, lessons, definitions, explanations, theorems, proofs, and diagnostic tests.

The function of the domain module is to act as a knowledge source and an evaluation reference. The knowledge source would generate assessments (problems and scenarios) and explanations (detailed discussions, guidance, and hints). The evaluation reference would be designed to provide a standard to evaluate the learner's performance (Far, 2000). For this implementation, we will only be concerned with the correctness of the current problem, regardless of previous history. The domain module will, in concert with the pedagogy module, create scenarios and questions with respect to the facts it knows to be true. The learner will be asked to solve problems on those scenarios. In our model, the learner will be required to show the intermediate steps he/she followed in obtaining his/her solution, hoping this way to avoid guessing the solution.

The communications module controls the actual interactions with the learner. The pedagogical module provides a model of the teaching process. It can control the content and the timing of the material to be presented. The expert module is a model of how someone skilled in a particular domain represents the knowledge and is sometimes modeled as part of the domain module. The domain knowledge module contains the information the tutor is teaching. Finally, the student module stores information about a specific learner. It can track past performance, current performance, and in some applications, the learner's mis-knowledge – or what they have not learned correctly. (Beck et al., 1996)

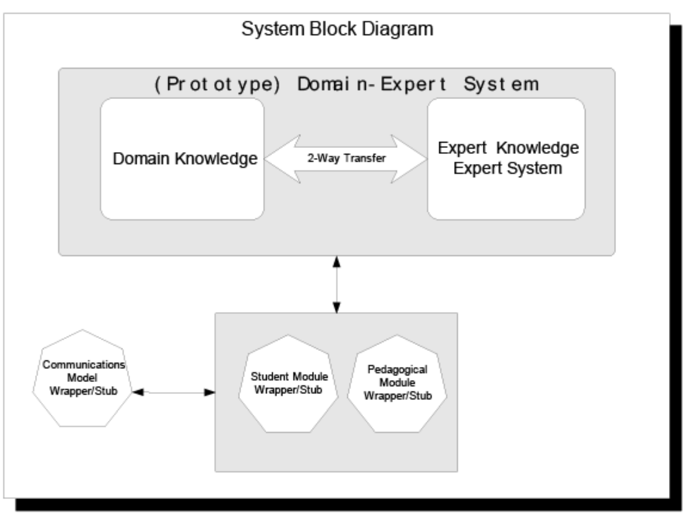
There are many ways to create the domain model. These can include production rules (if-else rule systems), ontologies (conceptual vocabularies), procedural and/or object oriented programming, and semantic networks.

The student model is more complex than the domain model. It is different for each student or group of students, depending on implementation. This model represents the whole of the learner. It includes their beliefs, level of competence, how fast and well they can learn, and where they have been and what they have mastered in the domain of the system. We defer the discussion and the development of the Student Module for later work.

APPLYING EXPERT DOMAIN E-LEARNING MODEL TO A LEARNING PROBLEM

In this section, we apply the e-learning model described in section 2 to a database normalization problem. Database concepts are taught at both the undergraduate and graduate level in most information systems, computer science, and computer engineering programs (Computing Curricula 2001). Anecdotal evidence shows that understanding and appreciating the full breadth of this topic is usually difficult for many beginning students. These students frequently employ the services of a peer tutor to master the concepts. Studies show that human tutoring provides significant achievement advantages.

Figure 1. Expert Domain E-Learning Model



Our proposed e-learning model is designed as a substitute to human tutors. Our goal is to provide the same one-on-one attention a student would receive from a peer tutor. It should be able to reason about both the problem domain (material under study) and the learner (the person(s) doing the studying) (Beck, Stern, & Haugsjaa, 1996). Since the system facilitates any time anywhere online learning, we expect the learner to gain factual knowledge on the chosen problem domain through the interactive and expert process built into the system.

We begin by developing the facts and rules our domain expert model is expected to "know". These include the underlying theory of database normalization, and an understanding of how problems in this domain are solved. Readers familiar with the relational databases and computation of closure may skip ahead to the section 3.3.

Relational Model

A relational database designer is required to achieve a higher form of normalization to minimize redundancy as well as insertion, deletion, and update anomalies ((Ullman & Widom, 1997), (Elmasri & Navathe, 2000)). This, in turn, requires the aspiring database designer to master the rules and complexity of the normalization process. One problem encountered in database normalization is the computation of the closure of n attributes in the database schema r(R) subject to a set of functional dependencies. In order to define the closure problem, we start by introducing the relational data model and associated terminology. This discussion is followed by an analysis of the closure problem. We assume that the reader is familiar with common terms such as relation, relation schema, and domains.

Assume X and Y are subsets of the schema R. A functional dependency $X \rightarrow Y$ on a relation r(R) is a rule expressing a one-to-one or a many-to-one relationship between the attributes of X and Y. The dependency $X \rightarrow Y$ holds in the schema R if for any two tuples t1 and t2 of r(R) which agree on attributes X=A1, A2, …, An, then they also agree in attribute Y. Therefore if t1[X]= t1[X] then it must be true that t1[Y]= t1[Y]. A dependency $X \rightarrow Y$ is nontrivial if X and Y are disjoint, i.e., no attributes from the right hand side appear on the left side. The dependency $X \rightarrow Y$ is minimal if no proper subset of X is also capable of functionally determining Y. We denote by F the set of functional dependencies that are specified on a relation schema R.

The set of all functional dependencies in F, as well as all those which can be derived from F, is called F+ the closure of F. The notation $F \Rightarrow X \rightarrow Y$ indicates that $X \rightarrow Y$ can be inferred from F. Consequently $X \rightarrow Y$ is a member of F+. In a similar way, the closure of a set of X attributes, denoted by X+, is the largest collection of attributes in R that can be inferred from X by application of the rules in F+.

Systematic reasoning on the construction of F+ requires the guidance provided by a set of axioms. An inference axiom is a valid rule indicating that if a relation r(R) satisfies some dependencies, then it must satisfy other equivalent dependencies. The following six statements – known as the Armstrong's axioms – are a complete set of inference rules for functional dependencies (Maier, 1983; Mannila & Räihä, 1992). In the following section A, X, Y, and Z are subsets of the schema R, and each of the axioms holds in r(R).

A1	Reflexivity	If $X \supseteq Y$ then $X \rightarrow Y$
A2	Augmentation	If $X \rightarrow Y$ then $XZ \rightarrow Y$
A3	Addition	If $X \rightarrow Y$ and $X \rightarrow Z$ then $X \rightarrow YZ$
A4	Projection	If $X \rightarrow YZ$ then $X \rightarrow Y$ and $X \rightarrow Z$
A5	Transitivity	If $X \rightarrow Y$ and $Y \rightarrow Z$ then $X \rightarrow !Z$
A6	Pseudo-transitivity	If $X \rightarrow Y$ and $YZ \rightarrow A$ then $XZ \rightarrow A$

Observe that if $F \Rightarrow X \rightarrow Y$ is true, then either $X \rightarrow Y$ is in F or it could be inferred from F. Inference is achieved by providing a proof in the form of a derivation sequence. A derivation sequence consists of an ordered sequence of statements in which a statement is either (a) an original fact in F, or (b) a new rule created by application of the A-axioms on any previous statement. Similarly if X is a group of attributes in R, then X+ (closure of X) could also be computed by a derivation sequence that ends at a point in which no more attributes could be added to the closure of X.

Finding the Closure of n Attributes in the Schema r(R)

Constructing the closure(X) requires an itemized proof showing that $F \Rightarrow X \rightarrow Y$ holds in r(R) and Y is the largest possible set of attributes in R that can be generated for the given X. Producing a derivation sequence to compute the closure of X under F is a non-deterministic problem. Roughly speaking, what is needed is a derivation sequence to show that $F \Rightarrow X \rightarrow Y$ holds in r(R) and Y is the largest possible set for the given X. A student must have an understanding of the axioms and the cleverness to provide a derivation sequence in which – in a constructive way – the right hand side of the rule is forced to grow up to its maximum. The mechanical process of producing the derivation sequence begins with the statement $X \rightarrow X$, however what follows could be many things. Let us illustrate with the following example: R=(ABC) and F={$A \rightarrow B$, $B \rightarrow C$}. Assume we are testing whether or not the rule $A \rightarrow AB$ holds in r(R). We could find a least three different derivation sequences to perform the computation, all different and all correct. Possible sequences are shown below:

Sequence 1:	1. $A \rightarrow A$	Reflexivity
	2. $A \rightarrow B$	Given
	3. $A \rightarrow AB$	Additivity 1, 2
Sequence 2:	1. $A \rightarrow A$	Reflexivity
	2. $A \rightarrow B$	Given
	3. $A \rightarrow AB$	Augmentation 1, 2
Sequence 3:	1. $A \rightarrow A$	Reflexivity
	2. $A \rightarrow B$	Given
	3. $B \rightarrow C$	Given
	4. $A \rightarrow C$	Transitivity
	5. $A \rightarrow AB$	Addition 1, 4, 2, 3

If we were to map out all the attributes and the application of each axiom to each attribute, we would see a tree-like structure. This structure may have many branches that either do not terminate or terminate in an endless loop. Only a relatively few of the branches would terminate in a solution. Any model that is created would need the intelligence to prune the branches that do not end with a possible solution.

Consider the proof provided by Sequence 3. If statements 4 and 5 are combined using A3 (Addition rule), a new line could be added stating $A \rightarrow ABC$. Observe that R=ABC, therefore no larger combination of attributes could be found and closure(A)=ABC.

A more constrained way of producing derivation sequences for closure(X) is based on the so-called B-Axioms (Maier, 1983; Mannila & Räihä, 1992). The B-rules are a complete set of three axioms having the same expressive power of the A-rules. The new rules are

B1	Reflexivity	$X \rightarrow X$
B2	Accumulation	If $X \rightarrow YZ$ and $Z \rightarrow A$ then $X \rightarrow YZA$
B3	Projectivity	If $X \rightarrow YZ$ then $X \rightarrow Y$ (and $X \rightarrow Z$)

From the previous example, the questions (1) does A'!AB follow from F, and (2) find closure(A) could be computed as illustrated in Figure 2.

At step 3, we observe that $A \rightarrow AB$ holds in r(R) and on step 5 we have to stop. All rules in F have already been used. The right-hand side ABC can not grow further, therefore closure(A)=ABC. Also, after step 5, we

Figure 2. A RAP Sequence for Closure(A)

Finding closure(A)	
1. $A \rightarrow A$	Reflexivity
2. $A \rightarrow B$	Given
3. $A \rightarrow AB$	Accumulation 1, 2
4. $B \rightarrow C$	Given
5. $A \rightarrow ABC$	Accumulation 3, 4

may use Projectivity to state that rules such as $A \rightarrow BC$, $A \rightarrow AC$ are valid in r under F. This approach is more focused, however the process is still non-deterministic and requires after each augmentation an examination of all attributes in the right-hand side of the growing X+ collection. We expect the domain expert model to be able to help students who are producing the derivation sequences using either set of inference axioms.

The e-learning system that we intend to create should provide assistance to the students in formalizing the systematic reasoning process. Observe that both problems, deciding whether a particular dependency follows from a collection of rules F, and computing the closure(X), are based on the same strategy.

Solving a Closure Problem with Proposed e-Learning Model

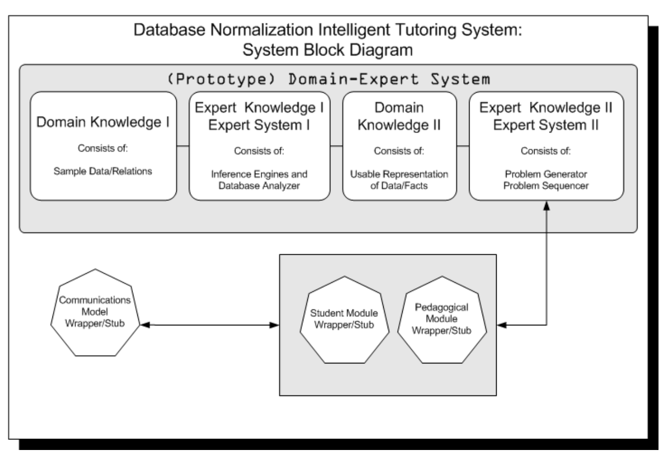
Using the general model from section two, we are proposing an implementation (Figure 3) that will aid learners in the self-study of the production of derivation sequences in the domain of database normalization. As before, the input for the model is provided by the instructor creating and populating the table to be analyzed using a relational database tool. The table is analyzed by the Expert Model. The model, using its knowledge of relational database design, provides facts to the Domain Knowledge Module. The facts include, but are not limited to: the relation schema, reduced set of functional dependencies, prime attributes, and keys and superkeys (Mannila & Räihä, 1992).

The problem generator and sequencer will be an expert system that will create a pseudo-random set of problems, of increasing difficulty, on the relation being studied. In addition to the problems, basic hint sequences will be developed to aid learners in finding the derivation sequence for the specified inference rules.

TEST PLAN

In order to accurately develop the problem generator, we first needed to see how different learners approach these types of problems. A number of subjects were asked to complete a given set of closure problems. Each subject was given the same problem set. As the subjects solved the problems, they were requested to say, out loud, what they are doing and thinking. The data was transcribed and analyzed. The collected

Figure 3. Applying the Model to Database Normalization



reasoning of the students as well as the approach known by the expert will be condensed as a sequence of statements from which an algorithm will be written. This algorithm will form the foundation for the domain-expert model. The model will be implemented in a prototype Intelligent Tutor System (ITS) application. This ITS application will be implemented as a wrapper, providing necessary support including the communications, pedagogy, and student models. After complete implementation of the model, we propose to test the model using student volunteers.

CONCLUSION

This paper develops a generalized e-learning model that would be used to provide pedagogical services for training students in solving complex problem. We describe one such complex problem: finding the closure of n attributes in the schema r(R) under a set of functional dependencies F.

The model is based on how learners approach the problem of solving the closure. In concert with a tutoring system that will be created in future work, this will allow the students to more closely ally their problem-solving skills with that of the domain expert and the known rules for solving the closure of a set of attributes.

REFERENCES

Beck, J. E., Stern, M. K., & Haugsjaa, M. (1996). Applications of AI In Education. ACM Crossroads, 3.1.

Computing Curricula 2001. (2001). Retrieved November 3, 2003, from http://www.computing.org/education/cc2001/final/chapter01.html

Conrad, K & TrainingLinks 2000. *Instructional design for web-based training*. Amherst: HRD Press.

DeRouin, R.E., Fritzsche, B.A., Salas, E. (2004). Optimizing e-learning: Research-based guidelines for learner-controlled training. Human Resource Management, 43(2-3), 147-162.

Deubel, P. (2003). Learning from Reflections - Issues in Building Quality Online Courses. Online Journal of Distance Learning Administration, VI(III). Retrieved from http://www.westga.edu/~distance/ojdla/fall63/deubel63.htm 09182004.

Elmasri, R., & Navathe, S. B. (2000). Fundamentals of Database Systems (Third ed.). Reading, Massachusetts: Addison-Wesley.

Engelbrecht, E. (2003). A look at e-learning models: investigating their value for developing an e-learning strategy. Progressio 25(2), 38-47

Far, B. H. (2000). TI-25 Advanced Lectures on Knowledge Engineering: Intelligent Tutoring System (Lecture Notes). Retrieved March 01, 2004, from http://www.enel.ucalgary.ca/People/far/Lectures/KE2/PDF/ke2-12.pdf

MacDonald, C. J., Stodel, E. J., Farres, L. G., Breithaupt, K., & Gabriel, M. A. (2001). The Demand-Driven Learning Model: A Framework for Web-Based Learning. *Internet and Higher Education*, 4(1), 9-30.

Maier, D. (1983). Theory of Relational Databases: Computer Science Press.

Mannila, H., & Räihä, K.-J. (1992). The Design of Relational Databases: Addison-Wesley.

Mason, R. (1998). Models of Online Courses. ALN Magazine, 2(2).

McArthur, D., Parker, A., & Giersch, S. (2003). Why Plan for E-Learning? *Planning for Higher Education*, *31*(4), 20-28.

Seufert, S., Lechner, U., Stanoevska, K. (2002). A Reference Model for Online Learning Communities. *International Journal of E-Learning*, January-March, 43-55.

Ullman, J. D., & Widom, J. (1997). A First Course in Database Systems. Upper Saddle River, New Jersey: Prentice Hall.