The community and its basic elements enable the process of teaching. For these teachers, the introduction of Educational Software Systems has been introduced into schools, teachers show increasing interest in educational software systems. For these teachers, developments in ICAI methods, introduction of Artificial Intelligence (AI) methods, introduction of computers into schools, and widespread use of computers by people of different vocation and age, made AI become a field of ever-increasing importance to education researches. The development of knowledge engineering, by its definition, methods and results becomes also directly applicable in education, based on the methods of knowledge engineering.

**INTRODUCTION**

Rapid development of computer technologies and Artificial Intelligence (AI) methods, introduction of computers into schools, and widespread use of computers by people of different vocation and age, made AI become a field of ever-increasing importance to education researches. The development of knowledge engineering, by its definition, methods and results becomes also directly applicable in education, whose goal is complete individualization of the educational process.

Intelligent Tutoring Systems (ITS) [Ta92] are very sophisticated software systems based on cognitive science and AI technology. According Kearsley [Wo90] ITSs are: "A computer program that uses AI technique to help a person learn." In Encyclopedia of AI [Ke87] the term ICAI - Intelligent computer-assisted instruction, which is also used for ITS, is summarized as follows: "Researchers who developed past ICAI programs have tended to be computer scientist rather than specialist in CAI... CAI specialists generally have backgrounds in training or education and tend to be largely ignorant of ICAI developments..." On the other hand, because several computers have been introduced into schools, teachers show increasing interest in educational software systems. For these teachers, however, ITS is the most difficult one in terms of grasping the total aspects. Of course they have a chance to read articles about ITS but what they can understand through the articles is confined to the abstract level of comprehension. Experience in using ITS directly through a system that transparentizes the inner mechanisms of ITS would be a powerful tool to aid them in comprehending it.

As research in ITS continues to produce more refined systems, the gap between the research community and educational community continues to widen. Educators' understanding, acceptance, and use of this research have been much slower than research progress. As this theory-application gap widens, it becomes more difficult for educators to participate in ITS research and application, and as result research becomes increasingly academic and unconnected to the pragmatic aspects of teaching and learning.

In designing and testing tutoring strategies there are several concerns. First, the experience of learning via ITS is so novel that neither teachers nor theorists can anticipate many important issues. Second, practicing teachers generally do not have well-articulated theories of learning. Third, relevant instructional and cognitive theories are not operationalized to a level easily implemented in a computer.

In consideration of the above trends and research issues, the goal of design the EduSof system is to design a conceptual framework for representing the objects, events, responses, reactions, and relationships involved in tutoring. Also, the second goal is to build a highly usable knowledge acquisition interface for rapid prototyping and easy creation, modification, deletion, and testing of both teaching concept and tutorial strategies. The third goal is if it is possible, to build a system that enables to teachers to build their lessons without any computer or AI specialists.

The paper describes one possible and convenient way of representing knowledge in educational processes based on the frame idea [Ni80], which was use for realization of the EduSof system, the system for generating lessons in almost all teaching subjects. By testing of the system, it has been concluded that this is especially suitable for different models of using in:
- mastering the subject and acquiring knowledge,
- testing the acquired knowledge,
- different other ways of knowledge testing, etc.

The EduSof programming system based upon such a way of presenting and representing of educational sequences, is implemented in the part of internal knowledge representation, knowledge acquisition and creation of lessons by the teacher, and the use for programmed learning and knowledge acquiring. The implementation has been carried out on IBM PC compatible computers under MS DOS operating system.

ON APPLICATIONS IN EDUCATION

Computers can be used in educational processes as a teaching aid, completely different from any other known teaching means. The specific thing about computers is that it enables a direct communication between computer and pupil. That means, that the computer could answer various questions asked by the pupil, leading him through process of learning, to improve pupils’ independence in learning, to individualize the teaching process, to advance through learning depending on the pupil's ability, etc.

As a condition for computer application in education, we assume the existence of high quality education software, based on teaching methods, didactic and pedagogical principles. Education software can be classified as follows [Je90]:

- Specialized educational software developed for unique fields (mathematics, chemistry, physics, etc.). They are highly specialized, rarely useful in the general purpose, their development is time consuming and expensive.
- General purpose educational software, which represents some tools for preparing and using knowledge in educational process, in almost all fields. Because of its generality, this kind of software can accept various sorts of knowledge representation.

Educational software EduSof is the general purpose educational software. As EduSof is intended to be well-designed, universal, and widely used programming tool for using in educational process, there appears the need for various ways of representing knowledge. EduSof knowledge consists of:

- some of textual information or textual knowledge represented by the means of text, and/or
- pictures (graphic representation of knowledge), as almost an obligatory element of all educational sequences, especially in the field of natural sciences, and/or
- simulations, because, in various fields experiments are a part of educational process, so there is a need for some simulation ability as a part of representing knowledge in the computer systems.

This means that suitable technique for representing knowledge in the computer’s memory must support handling:
- Basic knowledge - lessons, text, pictures, ideas, facts, etc.
- Additional information - to improve and help the using of basic knowledge.
- Reliable representation technique - representation formalism for different kind of information.
- Presentation and response mechanism - automatic demonstration of storing knowledge and mechanisms and various kinds of possible user reactions.

BASIC IDEA OF KNOWLEDGE DATABASE FOR EDUCATION

Among many known, different techniques and methods for knowledge representations [Pe88]:
- Functional model - frames, semantic or associative networks
- Logic model - rules, logical expression
- Object model – object-attribute value, triplets.

We use some hybrid knowledge representation methods, based on the object model of frame idea, modified with parameterizing semantic network technique in the part of the maintaining that knowledge.

In the research of knowledge representation within the AI a remarkable number of different AI tools that use frames [Fi85, Fo86] to structure knowledge have been developed. Parallel research in the field of designing databases tried to produce some tools for the development of semantics data models and object-oriented databases. These two types of system for representing the facts or knowledge into the computer seem to have much in common. They are structurally object-oriented and hierarchical. Both of them support inheritance of operations and relations within the objects of structured knowledge. They support memorizing external procedures or functions with the objects to which they are related.

There is no a widespread or standard definition of an object-oriented database, or on what facts they are based. In [Di86, Fr86] they are described with:

- Structurally, object-oriented databases illustrated the structure of a database using objects combined with their characteristics and properties, called attributes. In them, associations between the objects represented by their relationships are directly designed.
- Operationally and behavioral, object-oriented databases store operations (programs, procedures and/or functions) on objects with the data to which they are related and associated.

Frame systems were introductory created and conceived to describe a model of how humans established and arranged their abstractions, representations and ideas about the real world in their brain or mind. Frames have been widely used to represent and symbolize the prototypical objects in expert systems such as CENTAUR [Ai83], and they are used to structure data and rules of inference and conclusion on
knowledge, in some expert system environments and applications.

On the other hand we have several systems for representing information and some kind of knowledge that are currently developed. They are HyperText Systems, Authoring Systems, Computer Assisted Instruction Systems, and so on. All of them have in common that they are developed for memorizing and using various kind of information, which are connected without any didactic and pedagogical principles.

The COSTOC [Hu88] system tried to combine the methods from the Authoring, HyperText and Computer Assisted Instruction systems. The structure of the COSTOC system basic element, a lesson (Figure 1.), shows that the structure of knowledge used in them is static. Also, the method for using these lessons or knowledge is very dependent of the system, irrespective of the different paths provided through them.

Other direction is to design some generic shells for AI-based tutoring systems (SIIP - Macmillan et al.'s 1988, IDE - Russel et al.'s 1988, ID Expert - Merrill 1989, Byte-sized Tutor - Boner et al.'s 1986, PIXIE - Sleeman 1988, KAFITS - Murray and Woolf 1992, etc. These systems are intended to be used by educators or instructional experts who are not programmers. However, these systems are more focused on generality then usability. They do not clearly address the issues encountered when educators actually use these systems.

CONCEPTS OF AI METHODS IN EDUCATION

The power of using AI technology to build educational software result from the possibility of representing abstractions and models in the computer system. That means that is possible to encode some abstract entities such as "when the pupil is confused give more explanation," or "give topic overviews only for beginning level," or "give response for a right answer." To realize this degree of power and flexibility it is necessary to:

- represent models of the pupil and of tutorial expertise, and
- design a consistent, modular representational framework that support references to abstract objects such as "overview," "explanation" or "response."

ITS could be defined as intelligent, special-purpose systems with the feedback and the interface mechanism. ITSs are complex software products and represent the connection of several disciplines: education, AI, computer science, cognitive science, psychology, pedagogy, etc. The ITS [Cl83] consists of: Expert, Pupil, Tutor, Diagnostic and Inference module.

AN ITS DESIGN PROCESS

The design process of one lesson in some ITSs consists of the following sequential steps:

- **Qualifications** - summarizing main themes and models, which will be later expanded to the ideas.
- **Acquisition** - this process started with some kind of interviews with domain expert, proceeded with some kind of structuring that knowledge and methods. These included teaching a basic qualitative intuitive understanding of the relationships between some topics, the starting knowledge for the average pupil, common misconceptions and enumerating the important points of the lesson etc.
- **Scripts** - the conceptualization of the lesson, dividing (or structuring) the lesson in the important parts - concepts, and preparing the pictures and simulation screens and procedures.
- **Design of the topic network** - the lesson is expressed in the suitable knowledge representation scheme with the help of some software tool, i.e., that the relationships and the models of learning are connected with each topic of the lesson.
- **Designing tutoring strategies** - In the previous stage of design an ITS lesson ideas for "how to teach" were discussed and recorded in terms of informal rules, such as "to teach a composite topic teaches all of its parts" and "check that a topic is not already known before teaching it." The concepts used in these informal rules played a part in detecting the abstractions and properties needed to realize these tutoring rules in the computers.

One way to implement these rules is to use classical approach based on If-Then rules. In our investigation we found that the technique of parameterizing the action network of the semantic network of frames (this method is very similar to procedural flow charts) is more practically and manageable representational formalism then rules.

**KNOWLEDGE DATABASE IN EduSof**

EduSof system and appropriate knowledge database are designed as an ITS. The structure of knowledge used in them is dynamic (Figure 2.). The knowledge base is realized using the frame system. The teaching and examining methods for employing lessons or knowledge are widely open for lessons' creators (usually teachers), and are not dependent of the system.
EduSof has a great capacity of representing knowledge through teaching sequences in almost all fields of educational process. The textual information or knowledge is extended with graphics and simulation information or knowledge. So, EduSof is a tool, some kind of knowledge interpreter, easy to use, with a small need for experience in using the computer, assuming that making the programs for simulations would be a job of qualified people (professional programmers). In contrast to simulation procedures, the use of graphical presentation requires the knowledge of some of ready-made graphical packages (Dr Halo, Animator, and the like), as well as the use of some of tools for connecting the pictures drawn in the EduSof system.

In EduSof, a teaching sequence is organized as a list of semantically connected elements - Lessons. They consist of a list of Concepts. Concepts, as basic components of a teaching sequence, can be expressed by the means of Text, Picture, and/or Simulation. For each Concept, a list of Questions is given, used for testing the acquired knowledge connected to that Concept. For each Question, a list of Alternatives is given, as possible answers to that question, or the right Answer. For each Alternative, the Right Answer is provided. Also the information about What Next, in case that a right or wrong answer is chosen. So, the data structure used in EduSof can be represented as follows:

Lesson
Frame Start : Concept;
EndLesson.

ConceptName : Text;

Characterization : (Text ∨ Simulation ∨ Picture);

Frame Query : Question;
Frame Response : Answer;
Frame Explanation : Concept;
Frame WhatNext : (Concept ∨ Lesson ∨ EndLearning);
Frame Conclusion : Inference;
EndConcept;

Question
Characterization : (Text ∨ Simulation ∨ Picture);

Frame Query : Question;
Frame Response : Answer;
Frame Explanation : Concept;
Frame WhatNext : (Concept ∨ Lesson ∨ EndLearning);
EndQuestion;

Alternative
Characterization : (Text ∨ Simulation ∨ Picture);

Frame (Selection ∨ OpenAnswer ∨ Pairing ∨ ...);
Frame Response : Answer;
Frame Explanation : Concept;
Frame WhatNext : (Concept ∨ Lesson ∨ EndLearning ∨ Alternative);
Frame Effect : (Concept ∨ Question ∨ MoreDescription ∨ EndLearning ∨ Alternative);
Frame Conclusion : Inference;
EndAlternative;
EndAlternative;

EndLearning
Name : Text;

Frame Effect : (Concept ∨ Lesson ∨ ExitOfProgram)
EndEndLearning;

USAGE OF EduSof

EduSof, which is a natural extension of OSOF [Je88, Je90, Pu90], consists of three subsystems:

1) **Teacher**, the subsystem used for creating educational sequence, participation of all necessary knowledge for each concept and for creation of lesson structure and relations between concepts.

2) **Learn**, the subsystem used for learning and using lessons by the pupil, which are created with the aid of the Teacher module.

3) **Examine**, the subsystem used for testing knowledge acquired by using Learn module.

In Figure 3 is presented a graphical scheme of EduSof system organization. The system is organized in three separate modules, each of which is realized in such a way that enables its easy use even by the user with no much experience in using computers; the modules are the menu-driven and the user-friendly.

The creating module Teacher (Figure 4), serve for designing lessons by the teacher of the subject, module Learn (Figure 5) is a kind of interpreter of the knowledge introduced by the Teacher module, which is used by pupils during mastering the subject - matter and getting the pertinent knowledge. The third module can be used for different testing of pupils for marking.
The representation of the domain knowledge is object-oriented. The main object classes are lessons and the way of presentations. The concepts of the lesson represent units of knowledge that can be taught, remediated, summarized, etc. They are categorized according to knowledge type, for example: text, picture, simulation, more examples, and so on. The concepts have pointers, including various types of prerequisite, part-off and related-misconception links, to other concepts, forming the lesson network. They have pedagogical information such as summary, motivation, examples, tasks, etc., that point to presentation. The presentation or the array of questions or tasks, represent expository or inquisitor interactions with the pupil. They are composed of a task, such as a multiple choice question, or problem solving exercise, and an environment for doing the task, such as a picture or a simulation of some system. Presentation also contains the breadth of possibilities for responding to the pupil, such as hints, congratulations, elaborations of the answer, etc.

PARAMETERIZING SEMANTICS NETWORK

The method of parameterizing the semantic network allows many alternative tutoring strategies to be represented using a single learning module. Rather than defining new action procedure for each strategy, the generic action network is self-improving and realized the multiple ways of learning some lesson. In the next two Figures we show graphically the method of parameterizing the semantic network for the feedback mechanism, as an example:

CONCLUSION

The presented method for knowledge base appropriate for the educational is under development in the realization of the system EduSof in the Institute of Mathematics in Novi Sad. Further research will be in the direction of incorporating into EduSof the possibility of response recognition of spoken language, and resolving other AI problems, such as drawing conclusions, a possibility of computer generating of new lessons, etc., with the aim to make EduSof applicable in as much as possible of cases and subjects.

REFERENCES