In the name of GOD the compassionate the merciful

Subject Based Objects of Source and Curriculum Architecture, An Experience of Designing Computer Based Education Systems.

Ardavan Majidi
Department of Computer Engineering
Elm`o Sanat Iran University, Behshahr, Iran
majidi@pnu.ac.ir
www.pnu.ac.ir/majidi

Abstract

Computer aided education systems can used in wide area, Whereas CAE systems currently have been used only in research and laboratory domain, or in small limited applications and they have yet to make a substantial commercial impact. The development complexities of these systems, existing tools for developments, architecture of these systems, and cost of developing these systems, are some main problems and are reasons of this limitation. We believe that a well designed architecture with suitable tools based on this architecture, with a good interface that not require to any programming task in development of course material and curriculum, can decrease development complexities and costs, and may lead us to implement and use CAE systems in wide area as application systems in social learning environment. These motivated us to model an infrastructure of computer based education systems and design an architecture that make abstract the curriculum design, course material design, study and teaching from each other, that different people can work with system in own field of work and with own interfaces, abstractly. For this end we have designed SOSCA, The Subject based Objects of Source and Curriculum Architecture. This architecture prepared an abstract object oriented environment, based on learning subjects. For study and survey of SOSCA, we implement a prototype of a computer based education system that shows behavioral and architectural aspects of a CBE system, named MESBAH. However we are in first of way yet, all aspects that shown in MESBAH can be implemented by exist tools, knowledge and technologies. In this paper, after an overview on conceptual pivots of SOSCA and abstract explanation of these pivots, we will present the aspects of learning model and idea of tutoring scenario, and we will share our experience by presenting main structural aspects of this architecture.

Keywords: Computer Based Education, Architecture, Curriculum, Courseware, ITS, Educational Planning, Learner Model, Tutoring strategy, Tutoring Scenario, Object model, prototype.

1-Introduction

Computer aided education systems can used in wide area, and it is very useful if CAE application systems, becomes available to be used. Whereas CAE systems, currently have been grown and used only in research and laboratory domain, or in small limited applications. While intelligent learning environments have demonstrated their effectiveness as educational tools, they have yet to make a substantial commercial impact [Koedinger and others 1995 - Ritter 1997]. The development complexities of these systems, existing tools for developments, architecture of these systems, and cost of developing these systems, are some main problems and are reasons of this limitation [Wang and Chan 97- Nakabayashi and others 97 - Wong and Others 97].

We believe that a well designed architecture with suitable tools based on this architecture, can decrease development complexities and costs, and may lead us to implement and use CAE systems in wide area as application systems in social learning environment. It is important to increase capability of developing and organizing computer aided education systems in application domains in social learning environments. Social learning system are emerging learning environments that allow multiple students and agents to work at the same computer or across connected machines under different protocols of learning activities [Wang and Chan 97].

Social learning systems have attracted much attention because they socialize computer - assisted learning. Above problems make this hard. Our experience shows that many peoples tried to remove or decrease these problems, but we can not find any existing system or architecture that obviate these needs.

The lake of a good development tool is a big obstacle to the advancement of Computer Aided Education and Social Learning Systems [Wang and Chan 97]. Wang & Chan told that they could not found any existing development tool that fulfills the requirements of developing social learning systems in their past experience in researching breed of learning environments. They designed and implemented CAROL5, a simple general-purpose programming language to model and development learning systems. Also Ritter and Koedinger (1997) proposed one standard along these lines. They designed a set of primitives for communication between a tool and a tutor agent. They have presented an architecture for communication, cooperation and competition between multiple tutors agents. The important of that architecture is that it allows a system with complex behavior to be assembled out of simple components, each able to be developed without any knowledge of the behavior or even existence of the other. They hope that such an architecture will enable authors of educational systems with different approaches and interests to collaborate in building a complete system and will empower educators to assemble systems based on their own assessments of the value of the various components. Suthers and Jones (1997) also said that we must be able to add component functionality incrementally, and enable system to interoperate with commercial software and Internet resources.

We believe that one of the main problems of CBE systems development is its needs to working in programming levels, even if we use a suitable programming language for CAE development and with simple components [Majidi & Sadighi 99]. Work in programming level forcing us to constitute a team consists of programmer, science expert of specific field, educational planner and some other peoples in different scope. We know working of some people with different background in a team may increase complexity, cost of developing and deploying and time and decreased reliability and quality of development of such systems. Developing an effective instructional program requires careful and time-consuming task analysis, and true performance gains are seen only after several iterations of this process [Ritter 97]. Also this lead us to case base education systems that only present the education in single specific domain. It mean's for development of other domain we should constitute other team to do this. And from other aspects, our trying in share domains may be redundant. We know that exists many numbers of science domains that we need to CBE systems in them. So for development of these systems more than many cost and time required, its be require to exist many experience managers that lead this teams and experience programmers that implement programs with this much complexities, to obtain admissible reliability and quality. And we know we don't have these numbers of peoples with this level of experience. In the other hand, very of programming works in different case base education system, are same. So we have very redundant works in development.

These motivated us to model an infrastructure of computer based education systems that let the science expert and educational planner to design course material and curriculums, without need to any programmer. This lead us to design an architecture that make abstract the curriculum design, course material design, study and teaching from each other, that different people can work with system in own field of work and with own interfaces, abstractly. We try to design this architecture to prepare an infrastructure that multiple interfaces and different engines can manipulate different working aspects of education process with required tools, that powerful enough to develop such complex education systems [Majidi & Sadighi 99]. For this end, we have designed SOSCA; the Subject based Objects of Source and Curriculum Architecture. This architecture prepared an abstract object oriented environment, based on learning subjects. This architecture by organizing educational source and curriculum as objects, and defines them relations and specifications, and preparing some ways to design and edit curriculum, make possible to access course material trough curriculum structure with specific flexible organization. For this reason SOSCA used a model for organizing source of subjects. This model present subjects and its specifications in an object. The specifications are about status of subjects, relation to other subjects, kind of subject and so on. With polymorphism and generic class structure, SOSCA let the course designer to expand the system with defined organization.

For study and survey of SOSCA, we implement a prototype of a computer based education system that shows behavioral and architectural aspects of a CBE system, named MESBAH. MESBAH may clarify our mind and view of computer education systems and shows implemental aspects of these systems. However we are in first of way yet, all aspects that shown in MESBAH can implement by exist tools, knowledge and technologies. MESBAH shows that a CBE system, in an open university like as Payam'E Noor University, how can help the students, teachers and university organization, and how can lead this university to walk to become a virtual university. A virtual university is a university that for presenting educational services and official activities used minimum levels of physical environment and human resources.

In the following, after an overview on conceptual pivots of SOSCA and abstract explanation of these pivots like as subjects, subjects specification, object model and curriculum structure, we will present the aspects of

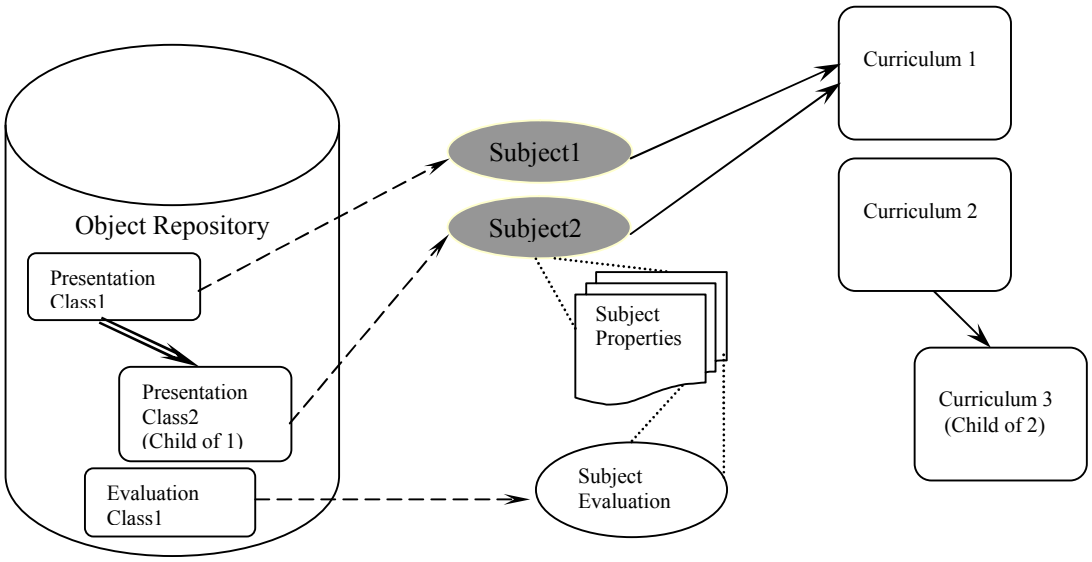

Figure 1: Subjects, Subject Properties, Curriculum and Object Repository in SOSCA.

learning model and idea of tutoring scenario, and we will share our experience by presenting main structural aspects of this architecture.

2- Conceptual Pivots of SOSCA

The SOSCA constructed on some conceptual pivots. These pivots, clarifies the structural lines of this architecture. The most important pivot is stand on subject. In this architecture the course material divide into atomic units called subject. However we suppose that a subject is an atomic unit, a subject may be consist of some other subjects. This let us to define a hierarchical structure from subjects. A subject contains a distinguishable and abstractable topic of learning material, and presents a specific subject of a specific domain of knowledge with some metaknowledge for managing and organizing subjects in a curriculum and course material and different sources.

As next pivot, each subject has its specification and properties. These properties specify some trait of subject to managing it when we want to construct a complex learning structure from subjects. These properties consist of information about subject general specifications, resources, subject evaluation information, subject relation to other subjects or resources, and subject metaknowledge specifications that help the agent for manage the relation of subjects in bigger structures.

As the next pivot, each subject, or each structures build on subjects, is an object, with all specifications of objects in object oriented design, like as inheritance, abstraction, polymorphism, encapsulation and so on. For example if a courseware subjects defined in system, it is encapsulated abstract entity, and author can define other subject as child of this subject, or prepare communication with other subject with specific messages. This lead to have generality and integrity in development of system and in curriculum editing. And with simplification of structure and naturalization, help to understanding structure by author in editing courseware. Objects makes possible to expand system, without any changing in system infrastructures, and this make possible to develop and maintain courseware structures in one hand, and reuse courseware components in other hand. However implementation of a complete theoretical object oriented agent has some complexities and problems, SOSCA prepare some main unambiguous aspect of object oriented design. One of the main important aspect that has used to increment development of a learning system by author, is presentation and evaluation classes. Each presentation class is a class of subjects that have same structures and used same method for learning. And each evaluation class is a class of evaluation methods that perform evaluation by a specific method. Author can define and drive each subject presentation from one of exists presentation classes, or he can define and drive each subject evaluation from one of exists evaluation classes. As shown in figure 1, presentation and evaluation classes take place in an object repository. Author can used this object repository and its classes, by driving an object from one of classes, or he can define one new class as child of other class and add it to object repository, or he can define new class as child of basic class. This let us expand the presentation or evaluation methods

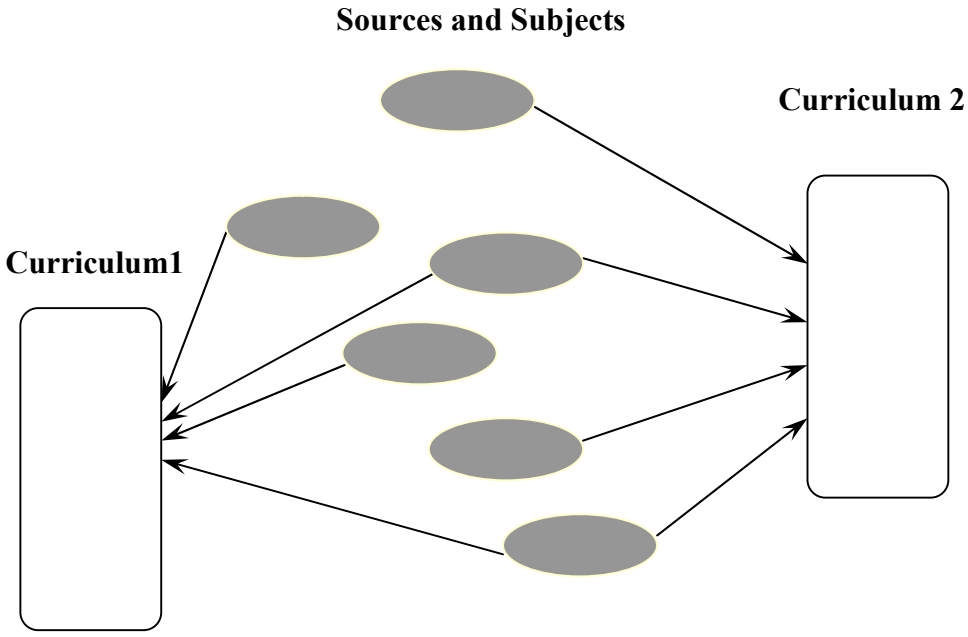

Figure 2: Sharing Sources and Subjects in different curriculums.

without any programming task. In none of these three kinds of using object repository author need to do his works in programming level. But a simple user-friendly userinterface helped him to perform this. As mentioned above, for reducing ambiguity in implementation and usage, SOSCA do not support multiple inheritance, But if find a way to removing this ambiguity, it may very useful in defining new learning classes. Also for simplify the use of object repository, we can define some template objects from exist classes, that can used by author in defining new subjects presentation or evaluation.

The structure of curriculum in a CBE system, is a main critical point. Chan (1992,1997) present the idea of curriculum-tree. However that the curriculum-tree not used directly in design of curriculum structure in SOSCA, but they have a cognition and have the same basic idea. The goal of developing curriculum-tree is to build a knowledge-based framework, which allows experts to construct their own learning companion systems for a complete tutoring course. The basic idea of curriculum-tree is that we can represent the abstraction of the curriculum as a curriculum-tree, a tree structure which organize the actual program of the learning activities according to the domain knowledge structure [Wang and Chan 97]. The curriculum tree construct is actually a 'divide and conquer' mechanism. It helps the author of the system to decompose the curriculum into small episodes, and then focus on the development of rule that are particular to an episode. Although curriculum-tree can largely reduce the complexity of developing social learning systems, it is still far from perfect. The major problem of curriculum-tree is that the framework in itself is too complex [Wang and Chan 97]. SOSCA used the same idea for curriculum structure. But for presenting curriculum, we use subjects and them specifications as shown in figure 1 and figure 2. In other words subjects specifications are main guidelines of building curriculum. Subject specifications when a group of subjects organized in a structure, used for simplify and managing it. In SOSCA a curriculum constructed by jointing subjects with their specifications in a tree or in a graph. To this point no problem appeared. Problem appear when we want to use this curriculum by an ITS (Intelligent Tutoring System) engine that it need a guideline for tutoring, based on this curriculum. This guideline as explained, prepared by presenting explicit knowledge. Representing and acquisition of this knowledge is main grate problem of developing an ITS and editing courseware. As it mentioned, this makes the framework complex. Without using explicit knowledge, how we can manage this framework of curriculum? Subject specification and properties used in this manner. When an author edit educational courseware, he or she can specify some main guidelines about this subject in its properties, that when this subject joint to other subjects in a curriculum, ITS engine can manage tutoring, based on this curriculum. This guideline parameter from other view is metaknowledge that by some specific parameter in properties may defined simply by author. Basic knowledge used by ITS engine is presented as some general education roles in system. They are no different in different courses. The representation of that knowledge is performed by combination of rolebased knowledge representation and semantic networks. Scenario based representation is also used in this manner.

Another pivot of SOSCA is stand on that education system can build from small components. The prospect of developing a complete learning environment can be daunting. One promising direction would be encourage the

development of small components of a full learning environment, so that any individual author's investment is limited [Ritter 97]. However coordinating the development of the varied components in a system can be a difficult task. Standards can be promoted which will allow components to be developed independently, with reasonable expectation that the components will interact properly when assembled [Ritter 97]. SOSCA by organizing subjects as abstract low dependent components, make possible this idea. Also to reduce the cost of materials prepared by developers, and to enable grater collaboration between users, representations of educational materials should be sharable between diverse applications across the Internet or in a social learning environment [Suthers and Jones 1997]. Suthers and Jones (1997) believed that Interoperability and reuse considerations suggest a "lowest common denominator" approach. They do not want to limit support for more advanced functionality such as domain-specific coaching. For address these concerns, they designed an architecture that places share resources. SOSCA for sharing resources has no direct strategy. It only let that different subjects from different sources, joint themselves within a specific course material, or in a curriculum. This makes possible to have a courseware consist of different parts that each part belongs to a specific various source. Each source may be one independent education system in Internet environment, or may be a part of a distributed education system. But SOSCA does not define how this various sources can communicate themselves and how they can share resources. By other words, we need another architecture for this end, to organize sharing and distribution of resources in Internet or in other same environments. We were designed that architecture and we will proceed to explain it in another time. SOSCA with that architecture prepared a framework for development of complete distributed education systems.

3- Learner Model and Tutoring Scenario

From other aspects, presentation of tutoring knowledge of subjects is very important. Zhang and Alem (1997) designed a simulation-based training system named ATEEG for Air Traffic Control (ATC) skill training. They believed that the curriculum planning in an ITS is carried out based on three components: curriculum formalism, learner model and tutoring strategies. The curriculum formalism represents target knowledge in an explicit or implicit structure. This specified in the curriculum structure that mentioned above. The learner model represents the system's perspective on the learner's state of knowledge. Most of the existing intelligence tutoring systems (ITS) teach students based on their domain Knowledge. Explicit cognitive skills of the students are generally not considered before delivering any lesson. Traditional user modeling system, mainly utilize user's knowledge about the application domain like user's expertise in that area and user's usage history with the system. The models build with such domain knowledge can be used with the specific application only, thus called Domain Dependent User Models (DDUM). The deficiency with such system is that they do not model the cognitive attributes or processes of the user, which are involved in all sort of cognitive activities taking place in humans. A cognitive model build with such atomic level cognitive processes (processes that remain same overall domains) behaves similarly irrespective of any domain. Such models will be applicable in virtually all domains, hence called Domain Independent User Models (DIUM) [Zia and Others 99]. SOSCA with constructing subjects properties and specifications, and with a similar structure for student specification, that shows students states in study, based on this curriculum and other related curriculums, has a simple limited Domain Independent User Models used stereotyping technique. A stereotype is a knowledge structure, which contains information about a group of student, a collection of attributes that often co-occur in people [Zia and others 99]. These attributes defined in student specification properties and subject properties.

After learner model, The tutoring strategy takes advantage of the structure the curriculum imposes on the knowledge and decides how to select and sequence the knowledge. The tutoring strategy applied by a certain tutoring scenario. A tutoring scenario specified sequence of pre-conditions and expected effects in learning process based on learning subjects and based on defined curriculum. A tutoring scenario employed a tutoring strategy, to rich the defined goal of this scenario in curriculum. Tutoring scenario's idea is near to idea of GCP (Global Curriculum planner) presented by Wong and Others (1997). GCP is capable of planning global strategies using information from the domain module. The domain module stores a set of teaching operations (teaching actions) with attached pre-conditions and expected effects. The pre-conditions and expected effects are presented as teaching goals. Tutoring scenario is constructed based on curriculum. By other words, when an author defined learning materials in independent subjects with they specifications and properties, and then he or she construct a curriculum based on these and other exist subjects, ITS can constructed a teaching scenario to lead this curriculum, based on structural points defined. By this scenario, ITS engine can able to manage learning process. This mean SOSCA prepare an environment that in it the scenario can constructed based on exist knowledge and defined subjects and curriculums. Student can pass lessons based on this scenario.

This used to study in defined way by educational programmer and science expert. But what happened if student want to study and walk in way out of this defined way? Can student do this? We say yes. As the same way that author defined this way or simpler, student also can define his ways as he like. This lead us to Exploration Study

Figure 3: A sample of using independent tools for presentation and MESBAH Interface.

or Open study or Free Education. In exploration study, student can select his way to study from exist ways. He or she can select subjects, lessons or courses that he can pass it, by attention to his last passed courses, his knowledge and his status in education system. Exploration study increased the student motivation and it may remove unessential steps of education. In exploration study, student can defined a goal point for this study, so all subjects appeared in his list can lead him to this goal. In other view of exploration study, we can use this for inquiry teaching. Inquiry teaching is a dialogue-based teaching style that forces students to articulate theories that are critical for a deep understanding of domain [Wong and others 1997]. By organizing subjects and curriculum and specify dependency properties of these subjects, we can have inquiry teaching in limitation that these dependency properties accurate and perfectly defined.

For applying tutoring scenario and inquiry teaching, Presentation of course material in teaching process is very important. Interactive using multimedia and virtual-reality as a medium in a computer aided learning systems is one of the important aspects of these systems [Shelbourn and others 99]. But it also very important that new way of presentation with new different tools may be used in a flexible long lifetime learning system. Because by fast evolution of computer and multimedia system, it is require to applied new methods and tools in a learning environment [Majidi 97]. So if we want to maintain a system in longer time, its architecture and design should no dependent to any specific presentation. For this end SOSCA let using any presentation tools and medium exists in operation system, by attachment of this tools to education system. The simplest way to using a presentation methods are using HTML or DHTML. And in more specific purpose XML is suitable. But it not means that we should use these as only way of presentation. In MESBAH we used PowerPoint presentation with it animation effects for this manner. Figure 3, shows a sample of MESBAH interface with a course material presentation.

MESBAH also for showing system architecture and its works, let the user to select a course from a tree that consist exist lessons that this student can taked in this time with attention to passed courses and student status. Selected lessons with it subjects appeared and presented step by step, and in each step, student evaluated by some question take placed in the subject specifications. All aspects exist in architecture shown in MESBAH in some forms and their fields. For example for showing domain independent student model in student specifiction form we have some fields to define some properties about student behavior in taking and passing courses. And in course definition specifictation forms we can define some specification of student behavioral aspects that when he or she wants to pass this course may be behaved. Finaly MESBAH simply shows how the organization of a

lessons and subjects presentation are performed by tutoring scenario and how we can define new coursewares and define their specifications and how we can have explorational study.

4-Conclusion

In this paper we have share our experience of designing SOSCA for developing Computer Based Education Systems. The most important pivot of SOSCA is that it is a subject base architecture. The next pivots are subject's properties, object model, curriculum structure and buildablity from small components.

We believe that separation of editing courseware from programming tasks with preparing suitable tools in object oriented environment and with good organization, can prepared a condition to use CAE systems as application systems in more than current limited domains. As the advantage of SOSCA we hope that this architecture can be able to: 1) abstract the tasks of different works like as designing courseware, editing course materials, editing curriculum and so on from each other and from programming task, 2) combine different resources by using different sources and preparing an infrastructure to share resources in a distributed environment, 3) increase development speed of courseware and decrease its costs, 4) make possible to exploration study or free education and 5) designing the courseware for interdisciplinary fields. And as an important specification of this architecture is that it is implementable by existing knowledge, resources and tools without any complex or ambiguous point.

SOSCA for sharing resources has no direct strategy and it only let that different subjects from different sources, joint themselves within a specific course material, or in a curriculum. It does not define how this various sources can communicate themselves and how they can share resources. We need another architecture for this end, to organize sharing and distribution of resources in Internet or in other same environments as further work. Mesbah Explanation is also available in Internet in: www.pnu.ac.ir/~majidi/mesbah.html

Acknowledgements

This study was performed in Payam'E Noor University. I should thanks to: Dr. Mohsen Sadiqui Moshkenani for his advice; Dr. Shahriar Pourazin for his advice; Dr. Saeed Abbassbandy for his help in performing MESBAH project; Eng. Yousof Besharati for his programming; Personals of computer administration of Payam'E Noor university and Kharazmi corporation for their help in performing MESBAH project.

References

[Majidi & Sadighi 99] A. Majidi, M. Sadighi Moshkenani. Organization of a Flexible Software Infrastructure - an Approach in Development of Computer Based Education Systems. Computer Report - 144, 1999.

[Zia and others 99] H. Zia, Q. S. Durrani, R. A. Farrakh, A. Riaz, F. Ahmed. CITS - Intelligent Tutoring System : A Domain Independent User Centered Curriculum Approach. IJCAI'99 Workshop on Learning About Users, 1999.

[Shelbourn and others 99] M. Shelbourn, G. Aouad, and M. Hoxley. Design a Computer -Aided Learning Application Using Multimedia to Train Inexperienced Building Surveyors in Building Pathology. Proceedings IEEE International Conference on Information Visualization, 1999.

[Majidi and Sadighi 98] A. Majidi, and M. Sadigui. Computer Aided Education in Iran's Environment. Conference of Training Technology, Allme Tabatabaee University, Tehran, Iran, 1998.

[Majidi 97] A. Majidi. Maintaining Balance to cope with the fast evolution of computer systems. Third international annual computer society of Irans Conference, Elm`o Sanat University, Tehran, Iran, 1997.

[Wang and Chan 97] W.C. Wang, and T.W. Chan. Experience of Designing an Agent - Oriented Programming Language for Developing Social Learning Systems. AI-ED 97 World Conference on Artificial Intelligence in Education, Kobe, Japan, 1997.

[Lester and others 97] J.C. Lester, S.A. Converses, B.A. Stone, S.E. Kahler, and S.T. Barlow. Animated Pedagogical agents and Problem -Solving Effectiveness: A large - Scale Empirical Evaluation. AI-ED 97 World Conference on Artificial Intelligence in Education, Kobe, Japan, 1997.

[Ritter 97] S. Ritter. Communication, Cooperation and Competition among Multiple Tutor Agents. AI-ED 97 World Conference on Artificial Intelligence in Education, Kobe, Japan, 1997.

[Nakabayashi and others 97] K. Nakabayashi, M. Maruyama, Y. Koike, Y. Kato, H. Touhei, and Y. Fukuhara. Architecture of an Intelligent Tutoring System on the WWW. AI-ED 97 World Conference on Artificial Intelligence in Education, Kobe, Japan, 1997.

[Wong and others 1997] L.H. Wong, C. Quek, and C.K. Looi. PADI-2: An Inquiry - Based Geography Tutor. AI-ED 97 World Conference on Artificial Intelligence in Education, Kobe, Japan, 1997.

[Suthers and Jones 1997] D. Suthers, D. Jones. An Architecture for intelligent Collaborative Education Systems. AI-ED 97 World Conference on Artificial Intelligence in Education, Kobe, Japan, 1997.

[Zhang and Alem 1997] D. M. Zhang, and L. Alem. From Task Sequence to Curriculum Planning. AI-ED 97 World Conference on Artificial Intelligence in Education, Kobe, Japan, 1997.

[Chen and Others 1996] L.S. Chen, P.W. Liu, K.Y. Chang, J.P. Chen, S.C. Chen, H.Hong, J.liu - Using Hypermedia in Computer - Aided Instruction. IEEE Computer Graphic and Applications, V16, may 1996.