Table of Contents Search Author Index

INTERACTIONS BETWEEN ELEARNING AND MULTIAGENTS

Aboubekeur Hamdi-Cherif *
Chafia Hamdi-Cherif (nee Kara-Mohammed)
King Saud University - KSU
Al-Gasseem Branch - College of Science
Department of Computer Science
PO Box 237 - 81999 Buraydah
Kingdom of Saudi Arabia
Email: cherif@ksu.edu.sa
elhamdi62@hotmail.com

* On leave of absence from Université Ferhat Abbas – UFAS
Faculty of Engineering
Department of Computer Science
19000 Setif
Algeria

Abstract

We study elearning and multiagents interactions. While extending our previous object-oriented experiences, we show how multiagent technologies can be applied to elearning settings. Thus we propose to describe an early attempt of bridging the gap between web-based learning and multiagents capable of learning from experience. The ultimate goal sought is the development of a fully-automated multiagent environment capable of assisting in the elaboration and delivery of highly-personalized educational material effectively for anyone, anywhere at any time while taking into account each elearner's personal profile and dynamic behavior during the elearning process. The targeted tangible result will be a complete multiagent system to be built under constraints, partially known at the outset and progressively specified during the analysis phase and constantly refined during all design stages. As far as design is concerned, Unified Modeling Language (UML) is used throughout for easy readability, reusability and maintenance. For the time being, and as far as this paper is concerned, the attempt is to concentrate on the interaction between two core fields namely elearning and multiagents. Prospectively, much effort is still required to meet the actual challenges so as to scale up to real-life problems of any significant complexity.

Key words
Advanced learning technology (ALT), e-learning *infostructure*, distance education, web-based instruction, multiagents, learning from experience.

1. Introduction

Over the last decade or so, elearning has become an important feature of the Internet culture, although still in the making. Broadly speaking, elearning is synonymous with electronic learning, delivered *via* a broad array of technologies such as television, video tape, intelligent stand-alone tutoring systems, computer-based training, and the Internet. We are interested in web-based instruction either synchronously, *i.e.* in real-time, or asynchronously. Indeed, the available communication tools are either synchronous such as on-line classroom, telephone, videoconference, shared on-line document, audio / video chat or asynchronous such as email, fax, answering machine, newsgroups, forums and newspapers. The structure of a typical elearning course is based on readings and exercises from simple quizzes to projects, on-line discussion, forum participation, live chat (audio and video), time commitment, and video-conferencing, amongst others. Specifically, distance education takes place when teacher(s) and student(s) are separated by physical distance and technology such as voice, video, data, and print, often in concert with face-to-face communication and used to bridge the instructional gap [Lau93].

Based on our previous experience in integrating different paradigms [Ham94-1], [Ham94-2], we intend to find a unifying framework within the area of elearning blending multiagents. While extending our previous experiences [Har02], [Man02], [Har04], [Man04], we show how emerging technologies can be applied in principle to elearning settings. Each individual learns in a different way and behaves differently when learning. Actual systems are poorly adaptable to individual behavior of the end-user. Therefore, a friendly-user multiagent-oriented environment is welcome to bridge the gap between user's behavioral features and pedagogical considerations, with helps on demand, while giving a living example of multiagents application to elearning. The multiagent approach is used mainly for its flexibility and autonomy. This last concept can be summed up as the system's adaptability to different and unpredictable situations and generating human-like decisions with no recourse to external help [Nwa96].

The paper presents the following structure. In the Section 2, we describe some of the prominent elearning experiences undertaken in academia and industry based on the so-called Advanced Learning Technologies (ALT). The multiagent approach to problem solving in general is discussed in Section 3. In Section 4, we describe a novel intellectual environment as the interaction between elearning and multiagents. Section 5 deals with architecture concerns. A conclusion reports the main positive aspects of the proposed approach and hints to the possible future extensions.

2. Elearning revisited

2.1 Academic elearning experiences

Nowadays, the advantages offered by elearning education are deeply rooted in the advanced methods of transmission of knowledge. It is widely-accepted that the primary benefits of elearning in general, and that of web-based instruction in particular, over traditional classroom learning can be summed up in its economical cost and its self-paced quality, its availability anytime, anywhere, for anyone [JALN00]. Numerous systems and web-sites are dedicated to elearning. We will only describe some experiences.

Top

a) The KSU-UFAS experience [Har02], [Man02], [Har04], [Har04].
The *UFAS-KSU project* is a joint project between Université Ferhat Abbas Setif, Algeria (UFAS), and King Saud University (KSU), Saudi Arabia. Our aim in this project is the extensive use of Unified Modeling Language (UML) for the development scalable distance education platforms. The core application is based on development of decision-support systems as applied to curricula development. One of the other important goals is to bring together on the same platform, instructors, course designers and course users. The aim is a better development, enhancement and use of instructional material. At the outset, our project was set out within the framework of the so-called *EUMEDIS* initiative, a large Euro-Mediterranean effort for the promotion of information technology in the 15 European countries and the 12 other Mediterranean countries [http://europa.eu.int/ISPO/intcoop/]. For research purposes, we had recourse to the focal point of the *EUMEDIS* initiative in Algeria, namely the "Centre des Etudes et Recherche en Information Scientifique et Technique" *CERIST* [www.cerist.dz]. The present paper is an extension of these seminal works.

b) Institute of Electrical and Electronic Engineers (IEEE) experience [www.ieee.org]
Not only dealing with elearning, the IEEE effort also addresses the issue of development of web-oriented applications for research and commerce. Several on-line courses and tutorials for certified technical training through commercial providers are offered by the IEEE. Moreover, IEEE has also moved to place its self-study courses on the Web, some entirely and others partially, in an effort to reposition its current array of products such as print-based self-study courses, videos, and CD-ROMs. One major challenge is to explore how IEEE and its Technical Societies can take advantage of the Web to promote their education programs as well as publications [Wie01]. As a result of this policy of promoting the dissemination of knowledge, IEEE - Computer Society has made available one hundred courses for all its members, completely free of charge. IEEE also actively participates in the standardization process through its Learning Technologies Standards Committee (http://lstc.ieee.org/wg12).

c) ACM (Association for Computing Machinery) Experience [www.acm.org].
Like IEEE, ACM is now making similar offerings of on-line courses. To date, approximately 200 courses are offered to ACM members.

d) The MIT Open Course Ware (OCW) experience [http://ocw.mit.edu].
The MIT OpenCourseWare (OCW) represents an upgradeable, database of courses. It intends to make all MIT courses available for the public *via* Internet. The OCW site opened in September 2002 and is freely available worldwide for non-commercial purposes such as research and education, providing a resource of some 50 initial courses, reaching 500 courses in September 2003 and aiming at 2,000 courses, by September 2007.

e) The All-learn experience [www.alllearn.org/].
Alllearn is a non-profit distance education company supported by three elite universities namely Oxford, Stanford and Yale. At its beginning, the company offered on-line courses to the *alumni* of these universities only. Since August 2002, *Alllearn* delivers on-line courses for the general public. The directories feature over 12,000 websites with 19 different subject areas. Now 50 courses in a dozen disciplines are available.

Top

f) The Fathom experience [www.columbia.edu ; www.fathom.com/]

Like *Alllearn,* Columbia University's for-profit elearning venture called *Fathom*, adopted a strategy of marketing eclectic courses to the public. So far, it has not been able to make profit yet. This means that academic excellence does necessarily imply commercial ability.

g) Other experiences

One of the earliest attempts to disseminate knowledge *via* electronic means such as televised programs remains the Open University in UK. The Open University was among the first higher education institutions to use a blended learning *i.e.* using more than one training space, usually face-to-face, instructor-led, and learning with web-based tools. The Open University study about elearners acceptance of web-based instruction shows that elearners still prefer using hard copy books to online technology especially when dealing with large amounts of course material. However, their study also shows that elearners make better use of course material when it is easily linked through a course web site [www.open.ac.uk]. Numerous institutions undertook similar Internet experiences worldwide.

2.2 Elearning solutions from IT Companies.

As a complement to the academic efforts described above, many elearning solutions are offered by Information Technology (IT) vendors. *U.S. News* and *brandon-hall.com* have teamed up to compile a directory of nearly 600 vendors that offer software, courses, tracking systems, and other electronic corporate training tools [www.usnews.com/usnews/biztech/elearning/]. We report the most prominent ones.

- IBM™'s elearning solution: the so-called IBM MindSpan™ Solution [www-3.ibm.com/mindspan]
- Sun™'s elearning solution: the so-called WebCT™ [www.sun.com].
- Blackboard™'s elearning solution: Blackboard 5™ [www.blackboard.com].
- Sun™ Center of Excellence. Sun™ and Blackboard™ teamed up in 1999 to establish the so-called Center of Excellence open in March 2002 [http://dotedu.wisconsin.edu].
- Macromedia elearning Suite™ from Macromedia™ [www.macromedia.com/software/elearningsuite/
- GamaLearn, one of the few Arab companies established in Dubai, UAE, offers elearning solutions [www.GamaLearn.com].

2.3 Limitations of traditional elearning environments

All the systems described above lack autonomy and have proved to be too monolithic to deal with the new expectations of individualized learning. Indeed, aspects such as data elearner personality, consistency and relevance become extremely important if one wants to have a truly-personalized learning. Besides, researchers in the educational field have shown that it is not possible to find a general strategy of teaching if we take into account human differences but it is rather probable to think that learning is an emergent result of rich and coherent dynamic interactions [Bal00]. That is why we consider new vistas for educational technology based on Top agents.

3. Multiagents

3.1 Brief definition

If we accept that intelligence is concerned with rational actions, we can define an intelligent agent, or simply agent, as a computational entity capable of perceiving its environment and that takes the best possible action in a given situation. An agent has the following basic properties:

- Acts on behalf of its users or other entities in autonomous fashion – a characteristics known as agency or autonomy.
- Behaves with some degree of proactivity and/or reactivity.
- Exhibits some degree of learning, cooperation and mobility.

When several agents work together on a single problem, we talk about multiagents [Bre98].

3.2 From agents to multiagents

It is useful to distinguish two principal categories of learning in multiagent systems *i.e.* centralized learning, or isolated learning, and decentralized learning, or interactive learning [Sen99]. Learning is said to be centralized if the learning process is executed in all its parts by a single agent and does not require any interaction with other agents. With that, centralized learning takes place through an agent completely independent of other agents—in conducting centralized learning the learner acts as if it were alone. Learning is said to be decentralized if several agents are engaged in the same learning process. This means that in decentralized learning, the activities constituting the learning process are executed by different agents. In contrast to centralized learning, decentralized learning relies on, or even requires the presence of several agents capable of carrying out particular activities [Nwa96].

In a multiagent system several centralized learners that try to obtain different or even the same learning goals may be active at the same time. Similarly, there may be several groups of agents that are involved in different decentralized learning processes. Moreover, the learning goals pursued by such groups may be different or identical. It is also important to note that a single agent may be involved in several centralized and/or distributed learning processes at the same time. Centralized and decentralized learning are best interpreted as two appearances of learning in multiagent systems that span a broad range of possible forms of learning [Mae94].

3.3 Software agents

3.3.1 What are they?

Although our work addresses the issue of multiagents applicability to elearning settings, we will concentrate on a specific type of agents namely software agents considered to be the most appropriate for the development of advanced human computer interface (HCI). The relatively short history of software agents can be traced back to the mid 1990's, a period of time that saw the interaction between HCI and artificial intelligence (AI) and lead to the new field of agent-based computing [Sho93], [Gen94]. Small-scale experiments with interface agents that learnt about their user [Lau97], [Lie97] and multiagent systems where simple agents interacted to achieve their goals [Woo95] dominated the research. Such agent systems, called software agents, were all grounded in the real world, using proven AI techniques to achieve concrete results. Users can delegate a task to a software agent rather than explicitly ordering the agent to perform it, usually *via* distributed AI (DAI) techniques [Sen99].

Top

3.3.2 Types of research in software agents
The research in this field is diverse and multifaceted. We report the most representative results.

- *Academic research*
  Small-scale projects are undertaken with non-brand names while medium-scale research is undertaken in some academic institutions such as Carnegie Mellon University (CMU), Massachusetts Institute of Technology (MIT), the University of London, *inter alia*. The number of universities actively pursuing agent technology is quite broad and the list is ever growing.
  Example: CMU's *visitor hosting system*. This system lies towards the "smart" end of the spectrum. In this system, "task-specific" and "information-specific" agents cooperate in order to create and manage a visitor's schedule to CMU [Syc95].

- *Applications in the tertiary*
  Many organizations now use agent technology.
  Example 1: *The Chronicle of Higher Education*, a weekly Higher Education newsletter uses agents to delivers personalized emails based on user's static choices, [www.chronicle.com].
  Example 2: Monster is a recruitment company that uses agents to deliver similar personalized emails [www.monster.ca].
- *Large Industrial applications*
  Large-scale research and development (R&D) projects are undertaken by large multinational companies such as Alcatel™, Apple™, AT&T™, BT™, Daimler-Benz™, DEC™, HP™, IBM™, Lotus™, Microsoft™, Oracle™, Sharp™, to name but a few. Obviously, the list is open. Clearly, these companies are by no means completely homogeneous, particularly if others such as Reuters™ and Dow Jones™ are added to the list. A short visit to the web sites of these companies shows their diversified interest in the agent technology. The scope of the applications being investigated and/or developed is arguably more impressive: it really does range from the classic *i.e.* nearly agent-free to the moderately smart ones.
  Example 1 : Lotus™ provides a scripting language in their Notes™ software which allows users to write their own individual scripts for managing emails, calendars, and set up meetings.
  Example 2 : Microsoft™ agents are used for orthographic correction, for adding new words to user dictionary, for search of files on hard disks, among other applications [Nwa96].

3.3.3 Further motivations for using Software agents
From the point of view of distributed artificial intelligence (DAI), software agents can be seen as a natural evolution from early multiagent systems, which in turn form one of three broad areas which fall under DAI, the other two being distributed problem solving and parallel AI. Hence, as with multiagent systems, they inherit many of DAI's motivations, goals and potential benefits. For example, thanks to distributed computing, software agents inherit DAI's modularity, speed, due to parallelism, and reliability, due to redundancy. Software agents also inherit AI properties such as operation at the knowledge level, easier maintenance, reusability and platform independence [Cha92].

3.4 Educational multiagents projects
Multiagents have been applied to as diverse fields as auction/market, entertainment, email filtering, expert assistance, matchmaking, meeting schedulers, news filtering, recommender systems and Web. Several projects implement learning systems based on multiagents

architectures. Some of them work on a generic platform of agents. Some educational systems include, but not limited to, the following.

- *JTS* is a web-based environment for learning Java language [Zap01] based on a CORBA platform and using Microsoft™ agents. In this environment, students have access to their student models and they are able to change it, in the case they do not agree with the information represented.
- *I-Help* [Vas99], a web-based application that allows students to locate human peers and artificial resources available in the environment to get help during learning activities. I-Help is an example of a large-scale multiagent learning environment [Vas01].
- *Pedagogical agents* Some interesting results have been achieved by regarding the student motivations [Joh00].
- *Companion agents* [Cha96] act sometimes as mediators [Con00a] within the learning process.
- *Tutor agents* [Rit97] are usually related to student modeling and didactic decision-making [Rit96].
- *Multiagents for distance education* [Web02]. Multiagents are used to teach plane geometry for secondary education.

Our aim is to enhance these contributions by adding a UML layer to the approach. UML has indeed grown to an industrial standard of software design because of its many advantages such as abstraction, inheritance, polymorphism, encapsulation, message sending, associations and aggregation [Con00b], [www.uml-zone.com]. The way how UML is used in elearning design has been reported in our previous works [Har02], [Man02], [Har04], [Man04]. Therefore it will not be described here.

4. ABED : A novel intellectual environment

4.1 Motivations: towards emergence of a novel technology

Our approach to elearning is an interactive synergy between traditional elearning and agents and represents an alternative methodology to actual elearning educational systems. We describe our Agent-Based intellectual Environment for eDucation (ABED). The chosen agent-oriented methodology brings several advantages to the development of educational applications. Indeed, this methodology deals well with applications involving different entities with unique personalities such as elearners with various learning habits, instructors, course designers with different cultural backgrounds, while integrating different components of software, ranging from ready-made programs to customized development tools. On the other hand, agent engineering, together with technologies of networking and telecommunications, bring powerful resources and opens new vistas for research and development of novel educational systems. Finally, the economical results are expected to be considerable for all levels of education, worldwide.

4.2 ABED Adopted strategy

In order to meet the requirements of the proposed environment, we follow the following strategy. Fig.1 shows the interactions within ABED.

ABED Strategy
1. Follow specifications set out by Learning Object Metadata (LOM) and Sharable Content Object Reference Model (SCORM) standards.
2. Study the process involved in design / development of non object-oriented agent-free

Top

systems [Ham94-1], [Ham94-2].

   2.1 Repeat the previous step for object-oriented (*via* UML) of agent-free systems [Har02], [Har04], [Man02], [Man04].

3. Study the process involved in design / development of non object-oriented agent-based systems.

   3.1 Repeat the previous step for object-oriented (*via* UML) of agent-based systems.

4. Study the process involved in "parametrized" systems (using agents on demand).

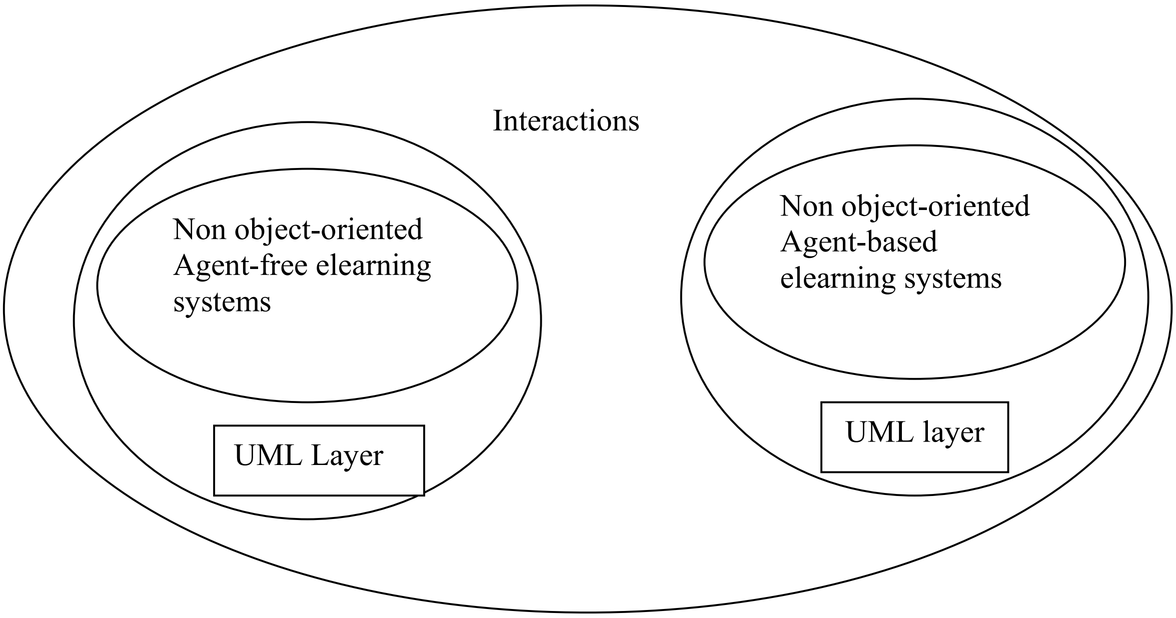

Fig.1 Interactions between elearning and multiagents in ABED

4.2 Compliance with standards

In our proposed strategy, the first step is to follow specifications set out by standards such as Learning Object Metadata (LOM) and Sharable Content Object Reference Model (SCORM).

<u>*Elearning standards*</u>

Like any environment, our intellectual environment ABED is supposed interact at one stage or the other with real-life systems like operating systems for instance. To be widely accepted, any environment has to comply with some adopted standards. For example, Internet as an environment cannot exist without standards like TCP/IP, HTTP, HTML, among others. Any scientific community needs its own standards and elearning community is no exception to the rule. Standards help the community achieve key goals for all parties involved such as tool designers, content producers, consumers, and tool vendors. Generally speaking, the purpose of elearning interoperability standards is to provide normalized data structures and communications protocols for elearning objects and cross-system workflows. These standards can be organized into general categories like metadata (data about data), content packaging, learner profile, learner registration and content communication. [Col02].

Top

Elearning Standardization Process
E-learning standards bodies like the Advanced Distributed Learning (ADL- www.adlnet.org/), a body working under the auspices of the American Department of Defense (DoD), the Aviation Industry CBT Committee (AICC-www.aicc.org) and IMS Global Learning Consortium (www.imsglobal.org) have accomplished tremendous work over the years. These groups' policies and standards have made it possible for end-users to run, for instance, a variety of content on any number of learning management systems (LMSs). However, two shortcomings have diminished the value of this work. First, the development of specifications, as well as the number of groups creating them, grew at a great pace. Second, many groups created standards that were engineering interoperability specifications best suited for developers and not for end-users. We will concentrate only on the issues addressed by the so-called Learning Object Metadata (LOM) and the Sharable Content Object Reference Model (SCORM).

Learning Object Metadata LOM *[http://ltsc.ieee.org/wg12/index.html]*
Learning Objects are defined as any entity, digital or non-digital, which can be used, re-used or referenced during technology-supported learning. Examples of Learning Objects include multimedia content, instructional content, learning objectives, instructional software and software tools, and persons, organizations, or events referenced during technology-supported learning. Examples of technology-supported learning include computer-based training systems, interactive learning environments, intelligent computer-aided instruction systems, distance learning systems, and collaborative learning environments. This standard is set out to specify the syntax and semantics of Learning Object Metadata, defined as the attributes required to fully/adequately describe a Learning Object.
For Learning Objects to be used, they must be found. This might be a truly challenging task if we consider a large distributed environment like the World Wide Web or a large intranet.
The LOM solution is to store objects (data) and their descriptions (metadata). LOM includes for instance information about the author, version, number, creation date, technical requirements, educational context and intent. We propose to apply multiagents in the use of metadata to support search, discovery, and retrieval of learning objects.

SCORM [www.adlnet.org/]
SCORM as a standard
The Sharable Content Object Reference Model (SCORM), devised by the ADL, aims to foster creation of reusable learning content as "instructional objects" principally destined to Web-based learning. First released in January 2000, SCORM continues to update and expand the scope of the specifications through cooperation with industry, government and academic participants. SCORM describes a technical framework by providing a harmonized set of guidelines, specification and standards for deploying elearning. At its simplest, it is a model that references a set of interrelated technical specifications and guidelines designed to meet high-level requirements for learning content and systems. By building upon existing Web standards and infrastructures, SCORM frees developers to focus on effective learning strategies.

SCORM Components
SCORM can be viewed as a collection of specifications and standards in the form of separate "books" in a growing library. Nearly all of the specifications and guidelines are taken from other organizations. These technical "books" are presently grouped under three main topics:

Top

- The so-called "Content Aggregation Model (CAM)" defines and treats the Sharable Content Objects (SCOs).
- The "Run-Time Environment (RTE)" deals with communication between content and LMSs.
- The "Sequencing and Navigation (SN)" introduced in 2004, covers the details of SCORM sequencing and navigation processes to include detailed coverage of how an LMS evaluates navigation requests and related activities.

Of course, the specificity of these books does not exclude possible overlap or mutual coverage.

SCORM interfacing
SCORM currently provides an Application Programming Interface (API) for communicating information about a learner's interaction with content objects, a defined data model for representing this information, a content packaging specification that enables interoperability of learning content, a standard set of metadata elements that can be used to describe learning content and a set of standard sequencing rules which can be applied to the organization of the learning content.

<u>ABED contributions to LOM and SCORM</u>
Because the standards do not specify how implementation is done, we propose to use the agent-oriented approach for implementation.
Example 1: use an agent mechanism to implement the guideline specified by LOM concerning the ability offered to learners or instructors to search, evaluate, acquire, and utilize Learning Objects.
Example 2: for SCORM, develop an agent-based mechanism to implement the so-called content aggregation, a process whereby the self-contained SCOs are assembled into larger and coherent packages of objects.

4.3 Overall operation in ABED

After complying with adopted LOM and SCORM standards, the second step in the ABED strategy is to study the process involved in design / development of non object-oriented agent-free systems, culminating with the use of object-oriented (*via* UML) of agent-free systems. Agents are introduced in the third and fourth steps. Because of the specificity of the present article, we will concentrate on these last two steps. Although the elements each agent perceives in the environment change from one agent to another, all agents have a similar general behavior, described in four main steps:

· <u>*Initialization*</u>
The same environment is shared by multiagents, by objects representing the solution constructed by the student, by objects describing the problem solved and simple objects representing candidate behavior. Agents are initialized in an inactive state, meaning that they cannot choose any action.

· <u>*First interaction*</u>
Agents perceive the environment. If an agent perceives in the environment the presence of elements represented, it becomes active. Since the agent is active, it is able to choose one set or more of candidate actions. On the other hand, if elements represented are not present in the environment the agent remains inactive.

· <u>*Next interactions*</u>

Top

Agents keep updating their states according to changes perceived in the environment. Action occurs in turns and active agents can choose eventually other relevant set of actions.

· Stabilization

When no significant changes are perceived in agent's states and actions, diagnosis is considered to be over. The state of an agent is conditioned by the presence of the elements perceived in the environment. An inactive agent has no right to choose actions. When an inactive agent becomes satisfied, it changes its state and becomes active. An active agent is able to influence other agents because of its activity. An active agent may become inactive again if at least one of the elements it represents is not present in the environment anymore.

4.4 Possible applications of ABED

In terms of applications, our approach is significant in the following scientific / technological areas with implicit economical results in the medium and long run. Applications address the following issues:

- To meet student needs and offer guidelines for better learning skills.
  Typical tasks include explicit guides for learning each specific material.
- To offer novel methods for improving interaction & feedback between students and instructors:
  Typical tasks include employee workload balancing, availability, absenteeism, extra lectures and infrastructure availability for each elearner.
- To improve design, planning & organization of courses and overcome computer imitations found in traditional elearning settings:
  Typical tasks include time table management, curriculum data, lecture frequency subject classification, linked courses, frequency, and schedules.
- To identify instructional development and the key phases of the process including design, development, evaluation, and revision of courseware:
  Typical tasks include topics, contents, course duration, teaching patterns, automatic paper settings, and assisted courseware elaboration.
- To set guidelines for agent-oriented evaluation of course delivery by specifying the types of evaluation:
  Typical tasks include but not limited to exam definition, schedule and frequency, subject allocation and classification, ranking, grades, promotion, absenteeism, and results dispatching.
- To allow elearning institution management:
  Typical tasks include managing trusts, trustees and founders, institution branches / classification, infrastructure, definition and classification.
- To point to common research questions such as agent-based learning *vs*. traditional elearning, the importance of multiagents, and cost *vs.* benefits.
  Typical tasks include contribute to a new area of research.

5. Architecture concerns

5.1 Possible design specifications orientations

Elearning applications are varied. They include but are not limited to:

- Computer-assisted instruction (CAI) uses the computer as a self-contained teaching machine to present individual lessons.

Top

- Computer-managed instruction (CMI) uses the computer to organize instruction and track student records and progress. The instruction itself need not be delivered *via* a computer, although CAI is often combined with CMI.
- Computer-mediated education (CME) describes computer applications that facilitate the delivery of instruction. Examples include electronic mail, fax, synchronous learning, asynchronous learning and other World-Wide Web applications, all based in traditional settings.
- We add here a novel area which we refer to as Autonomous Computer-Assisted Instruction (ACAI) based on agents emphasizing autonomy of decision-making in unpredictable settings. ABED is ACAI-based.

5.2 Chosen design specifications

As far as our task is concerned, we are concerned with the design of an Autonomous Computer-Assisted Instruction (ACAI) environment that uses agents as a core mechanism. Agents use a PAGE structure (percepts, actions, goals, and environment) [Rus95]. The idea is to generate actions from percepts in a given environment. We consider the following specific tasks destined to both asynchronous and synchronous modes.

- Specific tasks are concerned with the following:
  - ❖ To help in the definition of learning material with the actors involved (students, instructors, *ad hoc* specialists, staff), their ways of accessing to the system and their communication tools.
  - ❖ To define the type of tests and detailed content of courses concerned.
  - ❖ To manage changes (add/remove/update) in courses and actors properties.
  - ❖ To manage students' registration and pedagogical evolution.
- Design / develop / adapt associated tools and interfaces necessary for the task.
- Initially consider only a small, but upgradeable, range of activities to be undertaken by agents.

Upgrade activities from non object-oriented agent-free to object-oriented agent-based.

5.3 Overall tasks and operation

To design a system is to define its structure and its organization in terms of its components interaction. The basic structure of our architecture is composed of the following modules.

- *Eadministration*: a module that provides tools for the administrative management of all the actors such as elearners, instructors, and staff.
- *Ecourseware*: a module providing all tools necessary for courses taking, course elaboration and presentation, synchronously or asynchronously.
- Forum: module whereby the actors under the cover anonymity participate in the enhancement of learning process such as course content, delivery method, system operation, among others.
- Newsgroup module is provided for offline communication and accessible to all relevant actors.

All these subsystems are designed independently from course contents. This requirement implies that the contents structure has to be normalized so that it can be handled by the system's engine and in the same time has to be sufficiently flexible to maintain the themes diversity.

5.4 Examples of agents structure

Let us consider simple examples based on a PAGE agent structure as manipulated by ABED.

Agent type	*Percepts (inputs)*	*Actions (outputs)*	*Goals (performance measure)*	*Environment*	
				description	*properties*
Orthographic correction	Typed words	Print exercises, suggest corrections	Minimize typing errors or maximize student's test scores	Set of students	accessible, deterministic episodic, static, discrete
Eregistration (part of eadministration)	Student's actual level, prospective level.	Guide student in choosing possible pedagogical plan	Maximize student's satisfaction.	Set of students, set of staff	accessible, deterministic, nonepisodic, dynamic, discrete
Ecourseware	Student's actual level, prospective level.	Guide student in choosing selected coherent courses	Minimize course duration, maximize student's satisfaction.	Set of students, set of instructors	inaccessible, nondeterministic, nonepisodic, dynamic, discrete

For initialization purposes, forms are filled by all relevant actors to give the system a model of behavior of each one of them. Of course, each actor has right to change the model provided at any time. This initial model is used later by agents to decide which action to begin with and how to alter it when constructing a new search strategy.

6. CONCLUSION

This paper reports some prominent facets of ABED, a novel intellectual environment representing an interaction between elearning and multiagents. As result of this interaction a novel agent-oriented architecture for elearning is proposed. In addition to maintenance and modularity as key traditional concepts, autonomy is assured by multiagents which act in synergy for the benefit of the elearner considered here as a specific and unique entity. In our perspective, for each elearner and on the basis of some initial personalized information, a set of agents is dynamically allocated, according to the elearner's changing habits. Indeed, one of the challenge in an information-centered world, where human knowledge is the most valuable and scarcest commodity, it matters not only to make information available to people at any time, at any place, and in multiple forms, but also to reduce information overload by making information / knowledge relevant to the task-at-hand in accordance with the background of the users, their preferences, and habits. Through the mediation of our environment, elearners can enhance their ability to learn since they are indirectly helped during their entire search by invisible, yet very helpful, agents. Agents tune their behavior according to user's behavior, thus generating a human-like assistance during the elearning process.

Future work might include the use of automatic concept formation capable of clustering instances and their associated description into a hierarchy of categories. Ultimately, elearning is to be blended with more and more advanced methods of machine learning.

Top

References

[1] [Bal00] [1] Balacheff, N. "A modelling challenge: untangling learners' knowledge" *Journées Internationales d'Orsay sur les Sciences Cognitives: L'apprentissage, JIOSC2000*, Paris, 2000.

[2] [Bre98] Brenner, W., R. Zarnekow and H. Wittig. *"Intelligent software agents: foundations and applications"*. Springer Verlag, 1998.

[3] [Cha92] Chaib-draa, B., Moulin, B., Mandiau, R. & Millot, P. (1992), "Trends in distributed artificial intelligence", *Artificial Intelligence Review, Vol.* 6, pp. 35-66.

[4] [Cha96] Chan, T.W. "Learning companion systems, social learning systems, and intelligent virtual classroom" *Journal of Artificial Intelligence in Education*, Vol. 7, No. 2, pp. 125-159.

[5] [Con00a] Conati, C. Klawe, M. "Socially intelligent agents to improve the effectiveness of educational games" *Proceedings of AAAI Fall Symposium on Socially Intelligent Agents - The human in the loop*, 2000. [www.cs.ubc.ca/~conati/publications.html].

[6] [Con00b] Conallen, J. *"Building web applications with UML"*. Addison-Wesley. 2000.

[7] [Col02] Collier, G., Robson, R. "Elearning interoperability standards" Sun™ White Paper, Sun™ Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA, USA, 2002.

[8] [Gen94] Genesereth, M. R. & Ketchpel, S. P. "Software agents", *Communications of the ACM, Vol.* 37, No.7, pp. 48-53, 1994.

[9] [Ham94-1] Hamdi-Cherif, A. "A unifying framework for computer-aided engineering (CAE) with reference to computer-aided control system design (CACSD)" *Proceedings of the First Asian Control Conference (ASCC'94),* Vol. 1, pp. 45-48. Tokyo, Japan, 24-27 July 1994.

[10] [Ham94-2] Hamdi-Cherif, A. "The intellectual environment for a CACSD project" *Proceedings of the IEEE Conference on Control Applications (CCA'94).* pp.1365-1366.Glasgow, Scotland, 24-27 Aug. 1994.

[11] [Har02] Harbouche, K., A. Hamdi-Cherif "An object-oriented design for a distance education platform" *International E-Learning Symposium*, Dubai, United Arab Emirates, 12-13 May 2002.

[12] [Har02] Harbouche, K., A. Hamdi-Cherif "UML-based elearning architecture" *1st ElBaha Technical Meeting (BTM1),* El Baha, Saudi Arabia, pp.34-45, 2-5 May 2004.

[13] [JALN00] *Journal of Asynchronous Learning Networks* Volume 4, No. 2 - September 2000. Series of articles of interest: www.aln.org/alnweb/journal/jaln-vol4issue2.htm

[14] [Joh00] Johnson, W. L., Rickel, J.W.; Lester, J.C., "Animated pedagogical agents: face-to-face interaction in interactive learning environments" *International Journal of Artificial Intelligence in Education*, Vol. 11, pp.47-78, 2000.

[15] [Lau93] Laurillard. D. *"Rethinking university teaching – a framework for the effective use of educational technology"*, Routledge, London, 1993.

[16] [Lau97] Laurel, B. "Interface agents: metaphors with character" *Software Agents* (M. Bradshaw, ed.), MIT Press, 1997.

[17] [Lie97] Lieberman, H. "Autonomous interface agents" *Proceedings of the ACM Conference on Computers and Human Interface (CHI '97)*, 1997.

[18] [Mae94] Maes, P. "Agents that reduce work and information overload".*Communications of the ACM Vol*. 37 no. 7, 1994.

[19] [Man02] Mansouri, D., A. Hamdi-Cherif "An object-oriented decision support system (DSS) for curriculum contents elaboration" *International E-Learning Symposium*, Dubai, United Arab Emirates, 12-13 May 2002.

Top

[20] [Man02] Mansouri, D., A. Hamdi-Cherif "A decision support system (DSS) for courseware elaboration - UML-based design" *1st ElBaha Technical Meeting (BTM1),* El Baha, Saudi Arabia, pp.66-76, 2-5 May 2004.
[21] [Nwa96] Nwana, H., (1996) "Software agents: an overview". *The Knowledge Engineering Review*, Vol. 11, No. 3, pp. 205-244, 1996.
[22] [Rit96] Ritter, S., K.R. Koedinger, "An architecture for plug-in tutor agents" *Journal of Artificial Intelligence in Education,* Vol.7, pp. 315-347, 1996.
[23] [Rit97] Ritter, S. "Communication, cooperation and competition among multiple tutor agents" *Proceedings of the 8th World Conference of Artificial Intelligence in Education.* Du Boulay, B.; Mizoguchi, R. (Eds). IOS Press, 1997.
[24] [Rus95] Russel, S. and P. Norvig "*The intelligent agent book : artificial intelligence, a modern approach*", Chapter 2, Prentice Hall Series in Artificial Intelligence, Prentice Hall, USA, 1995.
[25] [Sen99] Sen, S. and G.Weiss In *"Multiagent Systems A Modern Approach to Distributed Artificial Intelligence",* Gerhard Weiss (ed.) The MIT Press Cambridge, Mass., 1999.
[26] [Sho93] Shoham, Y. (1993), "Agent-oriented programming", *Artificial Intelligence Vol.* 60 No. 1, pp. 51-92, 1993.
[27] [Syc95] Sycara, K., "Intelligent agents and the information revolution", *UNICOM Seminar on Intelligent Agents and their Business Applications*, London, pp. 143-159, 8-9 November 1995.
[28] [Vas99] Vassileva, J.; Greer, J.; McCalla, G.; Deters, R. "A Multiagent approach to the design of peer-help environments". *Proceedings of AIED'99,* Le Mans, France, pp. 38-45, 1999.
[29] [Vas01] Vassileva, J.; Deters, R.; Greer, J. McCalla, G.; Bull, S.; Kettel, L. "Lessons from deploying I-Help". *Proceedings of the Workshop Multi-Agent Architectures for Distributed Learning Environments, AIED' 2001*, San Antonio, TX, USA. pp. 3-11, 2001.
[30] [Wie01] Wiesner, P. "IEEE Societies: education is going for the Web" *Computer Applications in Power*, IEEE, Vol. 14 , No. 3, pp. 11 – 14, , July 2001.
[31] [Web02] Webber, C. & S. Pesty (2002). "A two-level multiagent architecture for a distance learning environment". url = "citeseer.ist.psu.edu/webber02twolevel.html" {Carine.Webber, Sylvie.Pesty}@imag.fr
[32] [Woo95] Wooldridge, M. J. Jennings N. R., "Intelligent agents: theory and practice". *The Knowledge Engineering Review, Vol.* 10, No. 2, 1995.
[33] [Zap01] Zapata-Rivera, J.D.; Greer, J. "SMODEL Server: Student Modelling in distributed multi-agent tutoring systems" *Artificial Intelligence in Education.* J.D.Moore, C.L.Redfield and W.L.Johnson (Eds.) IOS Press. pp.446-455, 2001.

Top