X-Learn: Integrating e-Learning, Resource Interoperability and Intelligent Tutoring Support

Juan C. Burguillo-Rial
Dep. of Telematics Engineering
E.T.S.E.T. 36200-Vigo. SPAIN
J.C.Burguillo@det.uvigo.es

Ernesto Vázquez-Guede
Dep. of Telematics Engineering
E.T.S.E.T. 36200-Vigo. SPAIN
ernesto@gist.det.uvigo.es

Martín Llamas-Nistal
Dep. of Telematics Engineering
E.T.S.E.T. 36200-Vigo. SPAIN
martin@det.uvigo.es

Abstract
Nowadays, the development of didactic resources to be shared among multiple partners is becoming a challenging task in the context of e-Learning. There is a clear need to use standardized educational notations to allow collaboration and interoperability among resources and tools developed in different institutions. Additionally, there is a need for smart learning environments that offer personal services with capabilities to learn, reason, have autonomy and be totally dynamic. In this paper, we present X-Learn: an Intelligent Educational System (IES) designed to simplify the elaboration of multimedia educational resources. To solve the interoperability problem, it uses packages proposed by the IMS organization and metadata proposed by the IEEE LTSC. It also includes questionnaires for auto-evaluation. Besides, X-Learn includes several intelligent agent assistants to improve its performance and to diminish some of the limitations of current e-learning environments.

Keywords
e-learning, standardization, intelligent software agents

INTRODUCTION
During last decade we have witnessed the success of the World Wide Web as a medium to deliver educational material. From the earlier Web-based educational systems to the present state of the art there has been an important evolution in a very short period of time. This evolution has taken advantage of improvements in the Web-based technology done in the recent past.

Nowadays, there is a need for smart learning environments that offer services to automate many of the time-consuming tasks that a teacher has to manage in a virtual course. Artificial Intelligence techniques have a role to play in this new kind of learning systems. The first Intelligent Tutoring Systems (ITS) appear during the eighties to personalize the learning material to the student. Nevertheless, it is in the nineties with the Web apparition, the lower computer's cost, the multimedia support and the Java language when arises with strength the paradigm of intelligent agents [16].

This paper starts presenting two experiences where our group has participated at an European level. These two experiences have provided a clear need for the educational standardization of tools and educational resources. Later, we present our current work centered on the environment X-Learn, which take advantage of the feedback gained from the last decade. X-Learn implements standardized educational notations to provide collaboration and interoperability among resources and tools developed in different institutions. Besides, X-Learn provides student intelligent tutoring and a tutor assistant using agent technology and Web support. These features convert X-Learn in an Intelligent Educational System (IES) oriented towards the net. Taken all together, we think that the X-Learn environment presented in the next paragraphs is a new brick to improve the design of e-learning systems for the near future.

STANDARDISATION
Information Society will have a strong impact on education and training systems in Europe. European degrees are attractive to universities as they provide added value to student curricula. At the same time, students can access to integral education from different institutions. In this framework several thematic networks have appeared at a European level to get a common synergy around these objectives. Our group has participated in several of these experiences:

- The European Thematic Network INEIT-MUCON [8] aimed at designing, developing and disseminating educational packages, for teaching EIE (Electrical and Information Engineering) in Higher Education. Our group has been participating in the Project together with 40 university institutions, having representatives from each country of the European Union. The main efforts were focused on the design and development of a basic set of didactic resources in the area of Electronics/Informatics, which can be disseminated through Internet to the European University community.

- The work previously released in the project INEIT-MUCOM received a strong impulse with the new proposal submitted to the European Community by its most active group of participants. The new thematic network (2000-2003), named THEIERE [15], gathers some of its objec-
tives from the previous experience and opens new collabora-
tion paths with new institutions from East European coun-
tries to include up to 80 universities. The THEIERE project
also aims at the challenging goal of getting a harmonization
of the curricula in EIE throughout Europe in order to facili-
tate the exchanges of students and teachers.

A clear drawback detected into these networks was the
presence of too much heterogeneity among currently avail-
able systems. That means that there is no way to reuse the
functionality implemented in a particular system by any
other because every one of them has their own proprietary
solutions. From the content point of view, it is also quite
troublesome to embody those pedagogical elements that
were thought to be used in a particular system in any other
different educational platform.

Much work has been done in the past and is being done
nowadays in the learning technology standardization field.
Two examples of institutions, which are gathering efforts in
the learning technologies standardization area, are the
IEEE’s LTSC [6] and the IMS project [7].

TOWARDS INTELLIGENT TUTORING SYSTEMS

Nowadays, although a long path has been walked and new
versions of e-learning environments include easy-to-use
Web authoring tools, most of them still offer passive ser-
dices. As a result, some instructors spend more time teach-
ing and tutoring students in a distance-learning course than
than teaching the same course in a classroom setting. For in-
fstance, the instructor is expected to regularly check the
courses they teach to evaluate the students’ progress by vis-
itng Web pages or didactic resources within the learning
system, to verify student progress and participation. This
process may include many time consuming activities like
monitoring the course forum, check if the students have
submitted their assignments, and regularly visiting the
course activity log to monitor the magnitude of students’
online activities. Many of these activities have to be per-
formed in parallel with the tutoring process itself, i.e., the
handling of many e-mail messages, a major time-consuming
operation for most instructors. At the same time, many tasks
from the student point of view, can also be automatically
managed by assistants to personalize the learning process.
For instance, personalized tutoring assistants may help the
student during the learning process with the intention to
adapt the learning material. At the same time, this tutoring
assistant may automate some tasks as: searching on the
Web, contacting with the class mates and providing the
teacher updated and accurate information about the student
progress.

Presently, there is a need for smart learning environments
that offer personal services with capabilities to learn, rea-
son, have autonomy, and be totally dynamic. The possibility
to use Artificial Intelligence techniques to provide com-
puter-aided support during the learning process appears
from a practical point of view in the eighties, where the first
Intelligent Tutoring Systems appear with the intention to
adapt and personalize the learning material to the student.
Nevertheless, it is in the nineties with the Web apparition,
the lower computer’s cost, the multimedia support and the
Java language when arises with strength the paradigm of
intelligent agents [16], the personal interfaces [13] and the
Intelligent Educational Systems [14]. These new set of
Educational Systems are called to adapt the offered services
to the multiple differences inherent to a global village audi-
ence. They have to be designed bearing in mind: didactical
considerations (how to teach), characteristics of the differ-
ent domains (what to teach) and personalized proposals
according to the end user (to whom we teach).

Concerning didactical considerations, there are several ap-
proaches depending on the chosen learning paradigm:

- Instructive learning: from this point of view, the
learning process centers the activity in the knowledge
transmission from the teacher to the student. The basic
model is a tutor adapted to and interacting with the student,
i.e., the learning strategy is completely conducted by the
teacher.

- Constructive learning: in this case, the learning
process is considered as a personal and active knowledge
construction. Actually, the student manages the learning
activity, performing her own sessions and fixing her own
learning objectives. The degree of freedom, in this situa-
tion, may be a negative factor when the amount of informa-
tion accessed can saturate the learner (for instance in docu-
ment exploration over the Internet). The role of the teacher
is to guide the student through the enormous knowledge
space.

- Collaborative learning: under this paradigm, the
student interacts with other students in a group where they
share explanations, justify reasoning, discuss solutions, etc.
This kind of learning requires a continuous motivation from
the teacher side. This approach complements the instructive
and constructive paradigms and differentiates the individu-
alized learning from the collaborative learning.

Unfortunately, the possibility to provide all this learning
paradigms and at the same time to manage all the inherent
complexity of modern e-learning environments is difficult
to implement. We need to integrate, in a distributed system,
the capability to deal with multiple sources of information
(the students) and the capability to manage this information
with artificial intelligence techniques.
We consider the use of intelligent agents [9] as a key tool to solve some of the problems related with the complexity of teaching duties in present distance learning environments. Since the last years, the concept of agent is becoming more and more important in both Artificial Intelligence and widespread Computer Science. The agent-oriented approach to design is based on decomposing problems in terms of autonomous agents that exhibit autonomous and sometimes intelligent behavior [16], and interact with each other in terms of higher-level protocols and languages.

When the agent model is applied to distributed system design, multi-agent systems or agent societies appear. Multi-agent systems are ensembles of agents, acting independently from each other to accomplish their own tasks. It has been shown that there are strong arguments in favor of agent-based software engineering [10]. For complex systems, agent-oriented decomposition is an effective way of partitioning the problem space, the key abstractions of the agent-oriented design attitude provide a natural approach to modeling, and agent orientation provides an appropriate philosophy for dealing with organizational relationships.

X-LEARN: STANDARDIZED EDUCATIONAL METADATA AND INTELLIGENT SUPPORT

Figure 1 shows the X-Learn system architecture whose main modules follow the client/server model using the programming language Java as the basic building block to provide services independent from any chosen software platform. Course tutors, course designers and students access the different tools of X-Learn using any commercial browser.

The right hand side of figure 1 shows the set of tools provided to the tutors and the course designers. There is exactly one component for every X-Learn module described previously. These components allow the creation, edition and management of the different courses and questionnaires. Besides, there is the possibility to develop agent-tutoring systems per course using a set of rules given by predefined templates. The Tutor Assistant also provides course assistance during the learning process. There is also a set of communication tools to simplify the interaction among tutors and learners.

The left hand side of figure 1 shows the student point of view. Student access is provided by any commercial browser. The student can follow the courses using the Course Viewer, can be automatically evaluated with X-Quest, has a Tutoring Agent to be assessed during the learning process and has also a pool of communication tools to interact within a group of students or with the tutor.

From our previous experience developing distance learning environments, we realize that it is very complex to maintain and update the changes in an environment with these features. Therefore, when designing the different modules we try to reduce their interactions using clearly specified interfaces in XML and following an engineering approach oriented to obtain a system composed by loosely-coupled components.
Course Authoring Tool

The course structure follows the IMS [7] package model, where the courses have a hierarchical tree structure as it is described in [7]. This format simplifies the definition and inclusion of new thematic units inside the course. It is possible to specify different course structures with several difficulty levels and the visualization of each level can be personalized according to the instructor criteria. Every resource or content in a course can be described with a metadata structure using the Learning Object Metadata (LOM) format specified by the IMS as it is described in [7]. The LOM format allows to reuse, classify and maintain the didactic resources, facilitating interoperability with other e-learning environments.

It is possible to specify access requirements (to have visited a previous list of resources or to have passed a set of questionnaires) for any particular resource and to define different learning paths (using different trees) depending on difficulty criteria. This information may be used, by the tutoring agent, to provide several versions of the same course, adapted to the knowledge skill of the student. This skill is evaluated through the on-line questionnaires, which are described afterwards. The course contents can be exported in different formats as HTML or LMML (Learning Material Mark-up Language) [12].

The Course Viewer

The learner may use the Course Viewer connecting to the X-Learn Web site. Once a course has been selected, a course tree structure is shown depending on the student skill level, together with the contents included in the selected resource. The learner can freely navigate in the course except when she desires to navigate through contents that have access requirements. In this case, first it is checked that the learner fulfils the minimum requirements specified to continue. In case that some requirements have not been fulfilled then a warning message appears and the access is blocked.

In the X-Learn server side, there is a Java servlet, which manages student authentication and sends to its clients the course contents. At the same time, this servlet checks the access requirements. Another task performed by this servlet is to store the student traces and the questionnaire evaluation data in the XML database. Those traces provide information for assessment to the human tutor, the teacher assistant and the tutoring agent. This information can also be used to generate statistics concerning the most visited resources of the course, those ones that are more difficult to understand, etc.

Questionnaire Tool: X-Quest

Questionnaires are a category of didactic resources that was not considered in the original specification of LMML. For these purposes we have developed a questionnaire tool named X-Quest [5] having these features:

- Test types: true/false, multiple choice, multiple answer, fill blank and their combinations.
- Questionnaires may contain multimedia resources: audio, video, Java programs, Flash animations, etc.
- It is possible to define feedback messages for every question/answer.

Figure 2 shows the friendly interface of X-Quest that allows to create questionnaires in a very simple way, but including advanced options as dividing questions in heterogeneous groups, establishing requirements for questions or marks that provide a great flexibility to evaluate the students.

Besides, the tool allows to follow the evolution of the students through an statistical module that shows information from two different perspectives: it allows to analyse the data obtained from a particular questionnaire or it can shows the data related with a particular student tracked by the system. Figure 3 shows a snapshot of the statistics interface.

All the information managed by X-Quest is stored in an XML database in the server and shared with the other modules of X-Learn. Besides, it is possible to import/export the questionnaires in an XML format compatible with the QTI specifications of IMS.
Typically, the student connects with the server using any commercial browser, fills the questionnaires and her answers are sent to the server where they will be processed and stored in the XML database. Afterwards, the server automatically sends the results to the student.

All the student information is accessible by the human tutors throughout the statistical module. Besides, other X-Learn components, as the Course Authoring Tool, the Course Viewer, the Tutor Assistant and the Tutoring Agent, may contact with a Java servlet that searches the information requested in the XML database.

**General and Communication Tools**

X-Learn have been benefited by the experience gained after developing the TeleMeeting system [2] and the Virtual Operating System [4], which managed on-line communications and off-line communications, respectively. Some of the applications designed and included in such systems has been adapted to X-Learn to provide support for groupware applications:

- **General Tools**: The system provides some general tools as an agenda, a calculator, an ASCII map, an user directory list and context-sensible help.

- **On-line Communications**: The environment provides real time tools to communicate among learners connected at the same time to X-Learn. Among the set of tools, we may cite a chat tool and a telebrowsing tool. With this telebrowsing tool, users can show presentations remotely. When a lecturer selects a new Web page with this tool, every learner's browser fetch that Web page automatically and present it to the user. This tool is very useful to complement other collaborative systems, for instance, multi-point presentations using videoconference [3].

- **Off-line Communications**: The off-line communication tools allow to exchange messages and documents among users not connected at the same time to X-Learn. One of these tools is a Shared File System that allows to share and to exchange documents among users in the same group. Another collaborative tool is the Message Forum, which allows to exchange messages among the users in a group and behaves similarly to the Internet News service. Another collaborative tool is the Message Forum, which allows to exchange messages among the users in a group and behaves similarly to the Internet News service. Finally, a Votation Tool has also been included. This tool offers the possibility to support discussions in groupware environments to know the participants' opinions. The inquire may offer different choices to allow the participants of a group to vote the preferred one.

**X-LEARN INTELLIGENT ASSISTANTS**

In this section we introduce the agent architecture used to manage the information provided by the teacher and the students to the learning system. The use of agent technology has revolutionised software technology and changed the concepts of classic software architectures [10]. The integration of agent and Web technologies is quite natural and prompts X-Learn as an Intelligent Educational System (IES) oriented towards the net.

In X-Learn there is one teacher assistant per course, while there are multiple instances of the tutoring agent to personalize the learning contents to every student. Besides, every tutoring agent is not always active, since students connect to the system at different times. Therefore, we decided to communicate the multi-agent system using an off-line scheme through the XML database.

The software agents implemented in X-Learn are servlets written in Java and we use JESS (Java Expert System Shell) [11] to analyse the input data and filter the information that is automatically managed. JESS uses forward chaining, which means that starting from a set of facts we get a set of new ones.

Clearly, the uses of JESS has been extraordinary useful and its integration with Java provides a perfect solution for the pursued objectives. This solution provides a set of agents very light, modular and easy to be adapted by the instructor.

In the e-learning context, we consider the use of Artificial Intelligence techniques as a complement to increase the quality of learning rather than a final solution.
A Teacher Assistant

Every course has a Teacher Assistant, which is a personal agent that may be configured by its owner, the human instructor. This configuration could include, for instance, the agent's level of autonomy to send overdue notices to students on behalf of the instructor, the language used in the body of the e-mail, the teacher tasks that will be automatically managed by the agent: tracking the students results, participation, interaction, etc. The human instructor must be constantly and dynamically informed of students' participation in a course and can assist a discouraged student before he or she drops out.

![Figure 4. Interactions of the Teacher Assistant with X-Learn](image)

Figure 4 presents the interactions among the Teacher Assistant with the rest of the modules. On the authoring side, the Teacher Assistant may provide direct information to the instructor about the information recently obtained from the XML server-side database. On the server side, there are servlets (see figure 1) that support these tasks to be performed by the Teacher Assistant. These servlets do not appear in figure 4 for the sake of clarity. When the Teacher Assistant is started, or any remarkable change happens in the XML database it informs the human teacher. Depending on the underlying expert system and the configuration introduced by the teacher, this agent automatically acts in three ways: modifying the XML database, using the Communication Tool Manager to send messages to the students or warning the instructor about new facts.

The action of modifying the XML database is the way to let permanent information to be used by the student-tutoring agents. The Teacher Assistant uses the Communication Tool Manager to send off-line messages to the students. Finally, the instructor is informed about data that can not be filtered and should be managed at a human level.

A Tutoring Agent

Concerning the learner side, the system provides a Tutoring Agent per student to help and supervise him/her during the learning process. The tutoring agent gets input data from the student traces generated by the course viewer and the evaluation data marks stored by X-Quest. From this data and a predefined set of rules, that can be adapted to every course, the tutoring agent can influence the learner to concentrate on those didactic aspects that need further revision.

![Figure 5. Interactions among the Tutoring Agent and X-Learn](image)

Figure 5 presents the interactions among the Tutoring Agent and the rest of the modules in the X-Learn environment. The Tutoring Agent interacts directly with the Teacher Assistant through the XML database storing data that should be processed on the authoring side by the instructor or the Teacher Assistant. Concerning the modules that appear in the student side, the Tutoring Agent interacts with the Course Viewer to get the student traces followed in the course. Besides, it also interacts with the X-Quest to obtain the marks achieved in the different questionnaires already corrected.

The Tutoring Agent assists students with specific learning needs, just like a human tutor or a classmate. It has some added features as a smart search engine, finding specific resources to solve learning needs for a student. Using student's preferences and interaction, the Tutoring Agent becomes more expert and useful as it provides more assistance to every particular student when it receives the feedback.
CONCLUSIONS

In this paper we have introduced X-Learn: an Intelligent Educational System (IES) that has been designed to simplify the elaboration of multimedia educational resources and to provide an added value with the use of agent technology to assist the teacher and the students through the Web.

To solve the interoperability problem, X-Learn uses packages proposed by the IMS organization and metadata according to LOM (proposed by the IEEE LTSC). It also includes LMML (Learning Material Mark-up Language) documents and X-Quest questionnaire formats. The use of standardized notation clearly enhances its interoperability and allows reusing the contents already designed.

Furthermore, X-Learn provides a set of intelligent agents to support both sides of the learning process. From the authoring side, it uses Artificial Intelligence techniques to provide software tutoring. Besides, the tutor can configure and automate many ordinary tasks. From the client side, the student is continually supervised and oriented to improve the learning effectiveness.

The uses of JESS together with the resource interoperability achieved through standardization makes X-Learn a modern e-Learning system that takes advantage of present day technologies.

X-Learn clearly supports instructive learning, which is the basic learning paradigm supported by most of nowadays e-learning tools. Concerning the collaborative learning approach, X-Learn supports the groupware paradigm through a set of on-line and off-line tools used to supervise the tasks proposed. Finally, the constructive learning paradigm can be obtained through the presentation of practical exercises to the learner that can be solved individually or in a group.

All these features (interoperability, agent technology, authoring, course displaying and questionnaire evaluation) convert X-Learn in an Intelligent Educational System (IES) oriented towards the Internet.

As further work, we are considering to expand the functionality of the software agents to provide a constructive and guided approach to solve practical exercises. The objective will be to include in any course a set of simulators to help the student to solve practical exercises guided by an expert system.

ACKNOWLEDGMENTS

The authors want to thank fruitful discussions maintained with Felix Buendía and José Vicente Benlloch in the context of the THEIERE thematic network and thank their valuable experience in the field of educational metadata. We also want to thank Juan José Vilas, Daniel Rodriguez and Patricia Eiras for their work in the implementation of X-Learn.

This work has been partially financed by the European Union through the project Thematic Harmonization in Electrical and Information EngineeRing in Europe (THEIERE). Project Nr. 10063-CP-1-2000-1-PT-ERASMUS-ETNE.
REFERENCES


