

From theoretical and experimental considerations to the Design of an adaptive distance-learning environment: the case of ARDOVIA

¹ Chihab Cherkaoui, ² Boubker Sbihi

¹ Laboratoire CERDIE, Université IBN ZOHR Ecole Nationale de Commerce et de Gestion BP.37/S Hay Essalam, Agadir, Morocco, ccherkaoui@yahoo.fr

² Ecole des sciences de l'information BP 6204 Agdal, Rabat, Morocco. +212 66 38 20 45 Bsbih@Esi.ac.ma

Abstract:

In most of the existing commercial platforms for distance education, the hold in account of individual differences between students is nearly non-existent or it often based on an instructor's presence. But, an instructor can sometimes be overflowed by the different electronic mails received. An efficient distance-learning environment must be able to adapt the content to all students by planning sequences of activities. The main goal of this communication is to show the different aspects bound to such a planning in the ARDOVIA environment.

Keywords: *Distance learning, Dynamic Planning, Multi-agent, Intelligent Tutoring Systems.*

1. Introduction

The Computer Environments of Human Training (CEHT)¹ based on advanced technology underwent a very appreciable evolution during these last years (Wenger, 1987), (Tchounikine, 2002). This evolution is not bound to the technological progress solely, it relates to cooperation between fundamental researchers and designers of systems, that one attends today deep modifications in the conception of the "learning" concept. Without wanting to go in details of the different theories of training, we underline merely here that the concept of learning begins to take new measurements as those bound to the concept of *space*, *time* and *totality* as underlined by Dennis and Lecleck (1995), who insist in the fact that the education must take place for all, all the time and everywhere, in individual and/or collective way.

One of the major consequences of this tremendous has been the rapid growth of technology-mediated distance learning at each level of education, and in particular the higher education. This interest becomes thus evidence that is not to demonstrate anymore. If it had been necessary to make it quickly, the variety and the wealth of themes proposed in the literature the different platforms proposed like: WebCt (WebCT, 2002), Claroline (Claroline, 2002), or Ganesha (Ganesha, 2002); as well as the diversity of experiences led in distance education would be the best proofs of it presumably. Each experience of distance education uses a particular meaning of the distance concept.

The most frequent conception used in the literature consists of the mixture of face-to-face and online learning, as varying the public, contents, subject matter and the used technological and educational tools (Hong et al., 2001). This issue, called Web-assisted course, is more important because it can clarify difficulties and avoid feelings of isolation that can arise in programs conducted purely at a distance (Modro et al., 2002). Another mode of delivery, called Web-based courses, utilises the Web as the sole delivery system and no face-to-face meeting is required. The theoretical foundations of these conceptions generally make reference to certain theories of the

¹ These environments are commonly called EIAH in French, CEHT is the translation. To simplify, we will refer to these as New Intelligent Tutoring Systems or ITS.

training as *constructivism* and/or *Social learning*. For the technical bases, we can underline the exploitation of most tools of Internet while using the Web pages, forums of discussion, electronic mailing, and Visio-conference. This is how most research done in this setting, put studies on the particular aspects of these environments as the possibility to increase the interactivity of Web pages. These studies most often correspond to the passage of static Web pages to the dynamic ones, where the user is not more merely a reader but a producer who can modify a document and control its behavior. It leads to the notion of active document possessing some really interactive components as applets, forms, while using standards as XML. Unfortunately, even if these researches permitted a big advanced in the domain of the interaction and the interfacings Man-machine, the different experiences conducted in one-line learning led to classical paradigms of training environments as the Computer-Assisted Instruction (CAI), that is as proposing itineraries pre-programmed itineraries, without taking in account the student's model to plan the individualized courses. The Learner it supposed to even manage his training, what puts the training forward by discovery, and complicate the task even though an instructor or co-learners presence, whose can reassure him and permit him to break with the isolation generated by the distance. But the instructor can sometimes be overflowed by the different demands he receives, notably in asynchronous mode, when it is about answering to a big number of electronic mails, in particular for the *big groups*, what is the case for the ARDOVIA project, as we will show it the next sections of this paper. Besides, if it can exists a big number of commercial platforms permitting to facilitate the creation and one-line course management as (WebCT, Claroline, Ganesha, ...) and as underlined in (Rasseneur et al., 2003; Faerber, 2003), these platforms, propose to students a "poor" individualized help.

For example, WebCT can only permit the student to see a simple historic of his progression including the number of connections done, the visited documents, or again the time of consultation of a Web page. Otherwise, for a student, the first quality of a teaching system is to be adaptable, and especially able to adjust teaching to his rhythm, and to his style of learning. The system must also focus on the different cognitive fluctuations and on the individual differences (Ohlsson, 1986). This adaptation cannot be done only if the system knows adequately the learner through a model of him (a learner's representation) that it constructs and that will be used by the system for matter to plan the different courses to propose.

It is therefore, when we discuss distance learning, essential to take in account the plural aspect of these knowledge systems, by permitting the *individualization* of the teaching, the *collaboration*, the instructor's *help*, but as the *guidance* of student activities while permitting a mixed initiative. These different aspects drive us therefore to review the different elements bound to distance learning in a larger conceptual setting as the Intelligent Tutoring Systems (ITS) Technology with Hypermedia concepts.

Us we will base our approach, on the one hand on the theoretical aspects of the problematic in question, notably those bound to the production of an adaptive distance learning environment (ADLE), that is capable, from a teaching objective, the model of the student, knowledge of the domain, but also of the educational knowledge, to plan an individualized sequence of activities (course or presentation, example, multiple-choice questions, exercises) for a student or a group of students. We will lean, on the other hand, on the experimentation that we led on the realization of a first prototype of the ARDOVIA environment (ARchitecture de production de DOcuments Virtuels Intelligents pour l'Apprentissage). We will note that the idea to combine theoretical and experimental aspects is very important, it requires a process of iterative conception with the numerous and controlled experimentation as underlined in (Tchounikine, 2002). This paper is organised in three parts. In the second section, we will mention briefly the methodology used in this research. The next section present the ARDOVIA environment: first its planning mechanism to deal with adaptability of the one-line courses. We will specifically show how the planning knowledge are represented and how these knowledge can be used to adapt learning for each student. Finally, we describe the current functionality and the implementation issues of the ARDOVIA environment.

2. Rationale of the research and Methodology

2.1. Research aims

This paper discusses the problems of developing WWW-based ITS and, in particular, the problem of combining the technology of recent ITS with Hypermedia concepts. The goal of the paper is to demonstrate how theoretical considerations of ITS systems and experimental considerations of an one-line course can guide the implementation of an adaptive distance learning system. The presentation is centered around the ARDOVIA project, which is expected to be used for distance learning to support algorithmic and Visual Basic programming course for a big group of students at the Agadir University. ARDOVIA is a built-in environment completing courses delivered in the face-to-face context. It offers a broad set of interactive models, including one-line course with adaptive sequencing, activities relating to "virtual classes" used in practical work in Visual Basic programming (Fig. 1). In the following of this paper, we refer to the whole of all these activities (face-to-face, one-line and Virtual Classes) by the ARDOVIA environment. The main characteristics of this research is to explore, analyze and represent data gathered in using an one-line course for Visual Basic programming in order to implement the adaptability required by the ARDOVIA system, as specified in the first steps of its conception. In particular, the questions driving the study were: How to characterize the necessary adaptability to the individualization of distance education? How to plan a course adapted to guarantee a follow-up individualized of every student without forgetting the importance of the social dimension as theorized by Vygotsky (1978) ? How we can avoid the isolation generated by distance learning (Belier, 2001), and how to answer efficiently and the most quickly possible to needs of a big group of students? How we can provide enhanced opportunities for converse, debate, and collaboration? In other words, what computer architecture will be capable to capture this adaptability? What type of planning process can we use? How can an instructional plan be determined? What are the components of an instructional plan? How this knowledge is represented? It is mainly this aspect of ARDOVIA the environment that we deal with in the following.

2.2. Methodology and data-gathering

2.2.1. Methodology

To explore what could be our contribution in the domains of the distance learning and ITS' systems, we achieved since 2001 a first prototype of the ARDOVIA environment. This environment was created to allow the 180 students of first academic year in higher education to complete courses delivered in face-to-face mode by a whole of on-line resources and activities. The system in question offers, on the one hand, some general information on the Visual Basic programming and algorithmic course, that permits to the different students, to get administrative information (sessions of practical works, plan of the course, of the guided works, of project statements...), and on the other hand, the system gives educational information, for example the statements of *practical works*, *guided works*, *past exams*, and the *correction*. Various links of educational Web sites is also offered. A library of exercises with the corrected exercises is proposed. Exercises we have actually in the system are of three types: Multiple-choice questions permitting to evaluate knowledge, exercises for a specific work at the level instruction in Visual Basic programming (Fig. 2), and others are having for role to allow students to combine the different instructions to write some longer computer programs. On the other hand, as specified above, we integrated to the environment ARDOVIA a tool (NetOpSchool, Fig. 1) allowing the teacher to follow and to manage practical activities in computers from his machine. In this Tool, several teachers can run each their virtual class on the computer Network. The teacher can monitor each activity of the connected students, he can for example:

- ü *Give a Demo* – The teacher can display the screen image of his computer or the screen of another student computer on the screen of selected student computers. He can for example, gives a demonstration for *saving files* in Microsoft Word.

- ü *Monitor Students* – The teacher can sequentially display and remote control the full-screen image of selected student computer;
- ü *Use Communication tools* – The teacher can start, stop and save a classic or audio chat forum session with selected Students

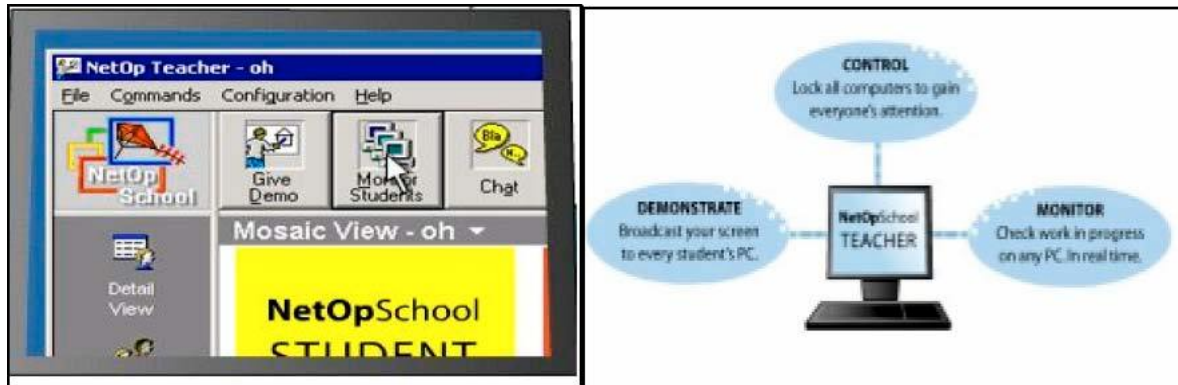


Figure 1. A Visualization of NetOpSchool

In this experimentation, in spite of the different assets that bring this learning environment, limits are obvious: they are caused by the distance and the solitude. Indeed, from the moment where student is alone facing an important mass of information, he will often have tendency to only select information that he already knows. Less we are competent, more we limit our training, and it is exactly the teacher's role to bring us where we would not simply have thought to go because we don't know the interest. A problematic requires an individual follow-up of every student, what doesn't constitute an easy task, especially when it is about big groups like in ARDOVIA. It is well obvious that the educational follow-up can be done of synchronous way through the "Chat" or asynchronous mode with the electronic mail. But the management of answers to the different messages received, is not a simple task, we counted during the year 2001 close to 3500 messages. The teacher or the instructor must answer every week to more than 170 messages. The load is of as much bigger during periods of exam. The first solution that we did consists in trying to regroup messages by theme and to try to answer there in the face-to-face course. There again, work is not simple. If in addition, when we are face to students with difficulties and that miss autonomy, the difficulty is to be obliged to plan for each student an individualized learning content.

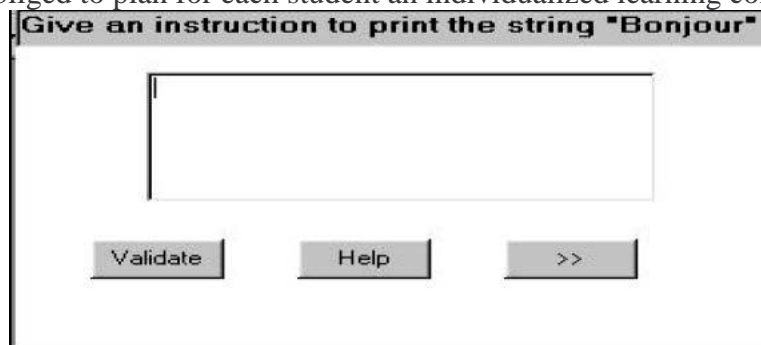


Figure 2. An instance of exercises at the instruction level in Visual Basic

All these considerations drove us to reconsider the ARDOVIA environment in terms of adaptability and the necessary individualization to an act of teaching and/or efficient training. In this context, the ITS technology is one of the most efficient way to individualize the teaching (Brusilovsky, 2000). Thereafter and in order to improve the quality of answer of the environment in question, we were brought the system more automated by integrating a component of dynamic didactic planning and a component of student modeling. This orientation essentially based itself on the first experimentation of the first prototype of ARDOVIA as a Web-assisted model, that has especially authorized to acquire different knowledge on students: their preferences, their ways to navigate and the consulted

items of the on-line course, to work in group and to share information, their behavior facing the different used media, their reactions, their questions about the courses (a basis of 6000 electronic messages has been cataloged and has been analyzed). All these elements are important to formalize different knowledge that we have used in the development of the ARDOVIA environment.

2.2.2. Data-Gathering

The different data gathered during the study comes from three sources of data described below. *Interviews* – Two individualized interviews with a representative number of students, before and after the period of electronic observation, provided the main source of data, particularly about their electronic interaction with the one-line course, their itineraries, mistakes committed in exercises, results of evaluations, sending mails to the instructor or the responsible teacher. These interviews show the different behaviors of students facing to the Man-machine interfacing, to the proposed course material, to given helps, to exercises; but it shows also the levels of acquirement and progression.

One line Interviews – To complete the classic interviews, we proceeded to the one-line quest (WebQuest) publication while trying to motivate to the maximum students to answer there. This procedure permitted us to compare the introverted data in the classic interview where the student is generally troubled by the device put in room (Camera, Teacher, and Instructor). *Electronic observation* – These observations have been led with the help of the *PageTuning* software of Cavalex Technology. This software permitted us to get all one set of measures on the behavior of students: past time on the on-line course, visited pages, the different reactions to news, reactions to the used media, etc. This information permitted us essentially of understanding the student's strengths and their special needs. This understanding leads us to complete the necessary expertise to the conception of the student modeling component, as well as the process of dynamic planning.

3. Some Aspects of the adaptive model of the ARDOVIA environment

Before describing the component permitting the requisite adaptability, we think that it is important to try to situate our work in relation to the different researches done in the domain. Indeed, the idea to produce systems capable to plan their behavior dynamically or to produce some "adaptive" systems in general is not new, it has already been identified in several domains, that it is in researches done in robotics (Fikes & Nilsson, 1971), in works achieved on the didactic control in the setting of the ITS (Futtersack & Labat, 1993; McCalla, 1992; Murray, 1990), in the WWW-based ITS (Brusilovsky, 2000), but also in the intelligent agent domain (Jennings et al., 1998; Maes, 1997). These different researches gave place to several models of planning on which we won't come back here because of the limitation the paper. In the setting of ARDOVIA, a problem of planning, consist to put a student from an identified initial state toward a final state that corresponds to the pedagogical objective. This passage is done by the execution of a plan of actions. According to this execution, the plan can be modified according to the different observations on the behavior of the student as well as of his assessment in relation to the waiting pedagogical objective. To formalize such a problem we must identify the different means of different knowledge representation and processes used in the progress of a session. It is more especially about the different entities of a didactic plan, and of processes of generation, execution and replanning. For that, we are inspired of the different works in Artificial Intelligence for the representation of the action as the STRIPS model (Fikes & Nilsson, 1971), but also of works on agents for the representation of the "rationality balance", that permits to establish the link between beliefs about the students, intentions of the system, and actions to accomplish the aimed objective. The goal of this section is to describe these different elements.

3.1. Entities of an instructional plan and their representation

An instructional plan is a structure of knowledge composed by four layers of which a simplified visualization is given in figure 3. Entities that constitute the different layers, by their nature and their own organization, are a matter for a theorizing of the teaching and a theorizing of the programming language training.

3.1.1. Pedagogical objectives

A pedagogical objective is a complex knowledge structure that corresponds to the problem to solve by the system that cannot analyze in detail in this article. Its complexity is bound, in part to the nature of the domain of programming language training aimed. Every task of training of a programming language puts in multiple expertise game that we will call factors.

An objective is represented therefore by a whole of factors. A factor is itself a quintuple with this form:

Fact :: = (<Fact_Name><Fact_Link> <Priority> < Eval_Modality >) (1) •

- ü *Fact_Name*: characterize the name of the factor while naming it. For example, factors that can intervene for a first group of sessions are: "F_Interface", "F_Procedures", "F_Events", "F_Objects", "F_Variables", "F-Printing ". In Visual Basic, the factor F-Interface, for example, consist to introduce the main elements of interfacing (TextBox, labels, buttons,...).
- ü *Fact_Link*: characterize links of dependence, inside the objective, between the factor and other factors of this objective. A link of type "Simultaneity" means for example that several factors can be reached of simultaneous way in one same activity proposed to the student.
- ü *Priority*: designate the order to accomplish the different factors. For example, for a beginning of training, the factor "F_Interface" will be more important than the "F_Procedures".
- ü *Eval_Modality*: constitutes an operator that characterizes the *intention* of the system in relation to the factor. In the present state of the system, we defined four operators of modality: M_Observation, M_Test, M_Have-acquire and M_Appropriation. So for example, M_Observation when it is assigned to a factor, the session will have the functionality to observe the state to student's knowledge. Beyond this representation of the objective essentially bound to the domain of training and in relation with the rational agent concept, several elements of the epistemic logic (Singh, 1994) and the modal logic have been introduced. In this context, an objective is defined by the following expression :

$$GA \equiv K \square A \square B \diamond A \quad (2)$$

Where A is a whole of factors. This equation requires for the system, that for having commitment in an objective, it is necessary to know (operator K abbreviation of Knowledge by opposition to "Believes" represented by the operator B, abbreviation of Believe) that the objective is not already reached and that it believes to the possibility to reach him. However we cannot know if the objective is reached without a consultation of the student model, it is one of most important elements of the dynamics of planning considered in ARDOVIA.

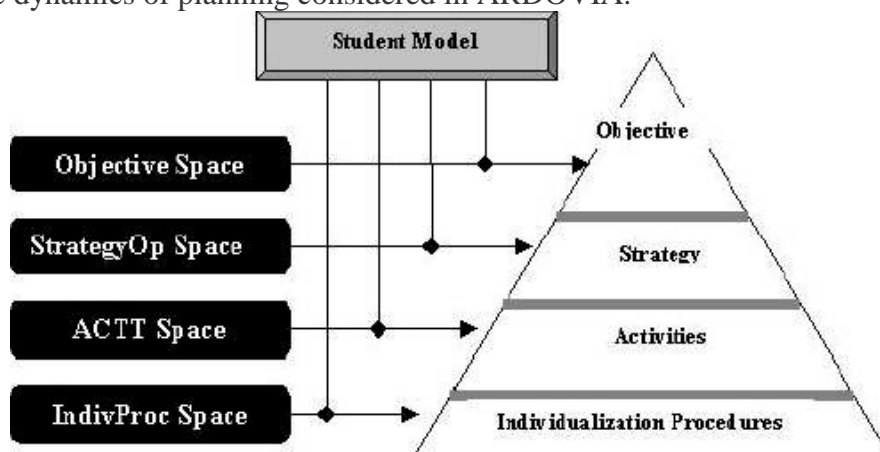


Figure 3–Structure of an instructional plan and its Generation.

3.1.2. Pedagogical operators and pedagogical strategy

The pedagogical operators represent types of actions available to the system in order to reach a pedagogical objective. For instance, we can mention four main operators: MIS (Mise en Situation), FIX (FIXation), CONT (CONTrôle) and APPR (APPRopriation).

- *The MIS operator* – It has for main role to allow students to acquire the basic concepts of algorithmic and programming. It is reached at a time in face-to-face context, but also by the navigation done with the plan of the on-line course proposed under shape of hypertext links, but also through the set of examples proposed on-line.
- *The fixing operator* – It is a stage in which the student begins to fix some ideas of courses proposed. It is reached at a time in the face-to-face context through practical work in a room equipped by machines and using the software NetOpSchool as we specified above, but as in on-line with a set of complementary exercises that every student must do.
- *The control operator* – This operator is also done of mixed way. In on-line context, the assessment of the student is continuous through exercises at the instruction level of programming, but also through the Multiple-choice questions. These assessments permit to support the student model, what allows the teacher to intervene through the electronic mails to reorient activities of students. In face-to-face mode, two continuous controls and an exam are proposed. Corrections are put at a time on the Web, but discussed also in face-to-face meetings.
- *The appropriation operator* – It takes place in face-to-face context when the teacher tried to show hiatuses of students as showing examples of the possible mistakes and bad conceptions of the students raised either on the Web or on exams. This stage is done with the help of explanations, examples and against-examples, but also with the help of returns toward the Web. The electronic mail allows the teacher to individualize the interactions with some students who have difficulties. We will note that in an instructional plan, an educational strategy is a sequence of operators; it is determined according to the modality of factors defined above. For example, the following rule:

IF M = ' M_Transmettre' and Current-Factor = ' F1 ' and Current- Objective= ' O1 '

Then pedagogical_strategy = (MIS(F1,O1); FIX(F1,O1); CONT(F1,O1)). (3)

Shows that if the modality of a factor F1 of the objective O1, then the educational strategy consists in putting in situation this factor, to fix it and to control its acquirement. It is this strategy that will permit to determine activity types to propose to the students.

3.1.3. Activities types

An activity type (ACTT) is a class of activities that represents an action susceptible to contribute to the resolution of the teaching problem represented by the chosen educational objective. In the setting of ARDOVIA, three classes of activity types exist: course documents, exercises and examples. And for each of these activity types, we have subclasses of activity. For example, for exercises we propose currently three types of exercises: the Multiple-choice questions permitting to evaluate the basic knowledge of students, exercises permitting a specific work at the level instruction in programming (Fig. 2), and others having for role to allow students to combine the different instructions to write some longer computer programs. The characterization of activity types is bound closely to the domain of the programming and the training in general; its representation is inspired in part of the classic diagram of the STRIPS model. Indeed, every activity has an effect waited in term of beliefs on knowledge the student. In the same way, each activity can only be proposed if pre-conditions are satisfied. The waited effects correspond to what we wish to make acquire for the students. These effects are represented in term of beliefs because of the uncertainty to know if effects are reached.

3.1.4. Individualization Procedures

Every activity type can be, according to different aspects (content, help, interfacing, etc.), adapted to a particular student. This adaptation can be made by determining individualization procedures of the ACTT, corresponding to each of these aspects. A following of individualization procedures, applied to an activity type will give place to a process of activity individualized ready to the execution. For example, for interfacing of activities of type Hypertext-document course (Fig. 4), we developed an individualization procedure that uses colors to facilitate the navigation; this issue is inspired from the ADAPT-TUTOR system (Czarkowski, 2000). The red color proposes an itinerary no counseled, whereas the green links proposes accessible pages to the student. Pages that have already been visited are coded in the blue color.

3.2. The Dynamics of Planning

The dynamics of the planning process is reached by the three big operative processes permitting the progress of a didactic session. We will present thus, successively, the process of *generation* of the didactic plan, the process of control of the *execution* and the *replanning* process.

3.2.1. Generation of an instructional plan

In the simplest case, this generation can be top-down generation (Fig. 3). The organized pair constituted by a *factor* and its *modality*, combined with other information coming in particular from the student's representation, determines a sequence of pedagogical operators. This sequence corresponds to a prototypical sequence of types of actions, which make possible to achieve, for this factor, the intention marked by the modality. For every pedagogical operator of this sequence, it is then an activity type that is searched for. Information associated to this ACTT drive to construct a student's representation waited correspondent in a state in the solution of the teaching problem. The requisite individualization here can be done to all levels of a didactic plan, even at the level of the choice of the objective. It is this process that is repeated, until the determination of an executable sequence of activities is found.

3.2.2. Execution of an instructional plan and replanning

An instructional plan having been determined thus, the system can precede the execution process. From an activity type, individualization procedures are applied effectively. We then obtain a structure of information that permits the efficient realization of the activity concerned. This structure is transmitted to a specific agent responsible of student interface management. This agent puts in work the activity in a «WorkShop» and manages interactions between the system and the student. In return, it addresses to the agent of execution an organized report of sequences of student's actions. These sequences of actions permit the system to assure the updating of the student's representation and to compare the deducted hypothetical representation, in relation to the waited representation. For example, EVT (click, Bvalidation,ACT1,1,t1), express the fact that the student has clicked on the validation button, in the activity ACT1. The number "1" designates the order of apparition of the event in the activity ACT1, and t1 designate a temporal reference of the apparition of the event. The process of replanning can be solicited in progress from different types of assessments of the development of the plan. This assessment can concern the partial progress of the current activity from the report of a complete activity or a group of activities. The replanification can carry on all layers of a plan. If student's actions of an activity show the necessity to interrupt this activity to modify some of its aspects (for example the available help or the remaining time for the activity), there will be a replanning process for the individualization procedures. This type of assessments can also leads to the modification of the pedagogical objective or the determination of a new one.

4. Description of the System Functionality and Implementation

4.1. General preview

When the student begins the session, the first screen he can access is a Portal of information related to the course, managed and automatically adapted by a group of technicians and administrative members. The Portal model serves to keep students informed as a whole, with timely and relevant information, as well as providing news, upcoming events, course links, and much more to all students, professors and tutors. The portal also contains the point of login for students, professors, and tutors (Fig. 4). Each student has to type his name and password, to create or recover his student model. If the input is valid, he has direct and unlimited access to course content and support tools.

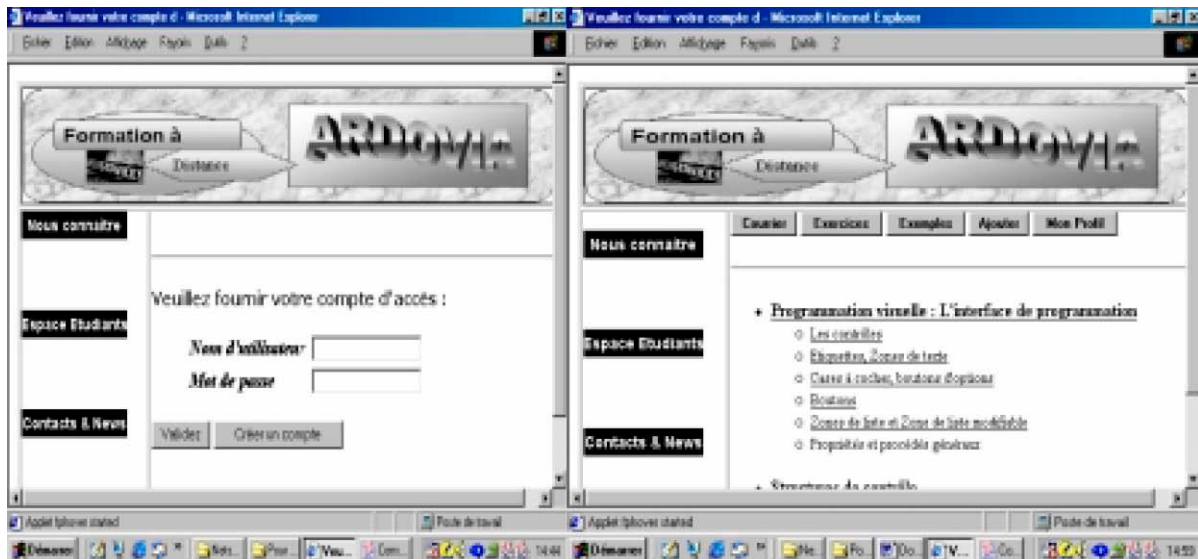


Figure 4. Sample visualization of the interfacings in ARDOVIA

If it is about a first session of connection, a pre-test is proposed to permit to establish a first model of the student. The proposed questions concern knowledge of the basis concepts of the algorithmic and programming. Then students can have access to an interfacing with content as the one of the figure (Fig. 4). The proposed buttons give to the student the possibility to access to the default plan of the course. The other buttons permit to give access to personal profiles, to an electronic mail possibility, to other Web links, to examples, or to exercises, and to the possibility to add examples or exercises that won't be active that after validation of the responsible teacher. A student can proceed in the environment according to two different modes. In the first mode, it is the system that chooses a progress adapted to the student's profile. Of alternative way, the student can come back to the default plan of the course and can choose other activities. This characteristic is comparable to certain systems as ADAPT-Tutor (Czarkowski, 2000). Colors, as we underlined above, are proposed in the different elements of the plan to facilitate the navigation. The red color proposes an itinerary no counsels, whereas the green proposes Web pages accessible to student. Pages that have already been visited are coded in blue. The

icons also allow the student to navigate according to his choices, what reinforces the autonomy in training.

We will note that other tools have been integrated such as:

- ü *Calendar* – gives students easy access to their course and personal events;
- ü *Notes* – an area where students can easily create and organize notes, which are linked to particular content pages;
- ü *Document sharing*– allows users (students, professors, and tutors) to share documents with specific groups, and facilitated participant collaboration;
- ü *Class list* – students and professors can identify who is online, send instant messages or emails.

4.2. Implementation

The architecture of ARDOVIA is conceived as a multi-agents system (Cherkaoui, 2003). Four Macro-agents are defined: the Planning agent, the student's modeler agent, the interfacing manager agent and the Expert of the domain agent. The Planning agent has for role to propose contents adapted to the profile of a given student (or student's model). This agent can save the different progress done by the student and propose advises on progress while putting in red links to validate as we already underlined above it. The Planning agent observes every action of the student that is captured by the interfacing agent, cooperate with the domain expert agent to analyze the answer of the student to identify some possible mistakes. If mistakes are detected, a suitable message is displayed, and help is proposed. It also proposes to the student's modeler agent to update the student's model.

For the implementation, we used at the server CGI scripts, the JAVA technology and the PROLOG language. Components permitting adaptability are also put on the server and arrange two bases of knowledge, the knowledge base called BCA that contains information on the different interactions of the student and the knowledge base named BCR that contains resources (Exercises, Course, links). Each of these resources brings has potential of reusability, generatively, adaptability. Forms of interactions are coded with the help of the HTML and the JavaScript code that play the role of event manager. The mechanism of the Cookies is also used to capture interactions of the student.

5. Conclusion

To summary different presented points above, we hope to have put a first problem draft bound to the conception and the development of adaptive environment for distance education. Among the essential points that we treated, we can more especially keep the different aspects bound to the dynamic planning, whose objective is to permit a follow-up individualized of every learning. We think that ARDOVIA reached of brief way the fixed objectives: it is currently operational. We have, indeed, recorded that on average 50% of students use it. We can note also an explosion that reaches the 95% of students during periods of exams, what generates a certain slowness of the system, however. For the resources of the system, we can note 400 exercises to the whole of which 50 proposed by the teacher and the remainder by apprenantses themselves. It is certain that several aspects of this problematic remain to develop to permit the conception of new CEHT that will be complete again, capable to be integrated in an institutional cursus. We can note certain problems bound to the automatic planning in particular and notably to the student's representation which is currently rudimentary enough, we think to improve it therefore while introducing the probabilistic models. We finally underline, that the present implmentation endures a slowness, probably owed to the CGI scripts, and that we intend to make migrate toward the PHP language, faster to the execution and easy to maintain.

6. Bibliography

1. Belier, S. (2001). *Le e-Learning*, Editions Liaisons.
2. Brusilovsky, P., (2000). Adaptive Hypermedia: From Intelligent Tutoring Systems to Web-based Education, *ITS'00*, Vol.6.
3. Cherkaoui, C. (2001). Une démarche multi-acteurs d'ingénierie de la formation, *3ème Congrès du RAPES*, pp. 57-58, Maroc.
4. Cherkaoui, C. (2003). Vers un rapprochement des Environnements Informatiques d'Apprentissage Humain et la FOAD", *SETIT'2003*, pp.122-132, 2003.
5. Cherkaoui, C., Chambreuil, M., Gaguët, L. (1997). Aspects de la planification didactique : étude dans le cadre d'un environnement d'aide à l'apprentissage de la lecture, *Sciences et techniques éducatives*, Vol.4, N°3, pp. 257-297, 1997.
6. Czarkowski, M. , KAY, J. (2000). Bringing Scrutability to Adaptive Hypertext Teaching, *ITS'00*, Vol.6, pp.423-431, 2000.
7. Denis, B., Leclercq, D. (1995). Apprentissage et multimédia, *Actes de la journée d'Information sur le multimédia*, Presses Universitaires de Namur, pp.119-141, 1995.
8. Faerber, R. (2003). Groupements, processus pédagogiques et quelques contraintes liés à un environnement virtuel d'apprentissage, *EIAH'2003*, pp. 199-210, Strasbourg.
9. Fikes, R.E., Nilsson, N.J. (1971). STRIPS : A new Approach to the Application of Theorem Proving to Problem Solving. *Artificial Intelligence*, vol.2, pp.189-208, 1971.
10. Futersack, M. (1990). *Quiz : une architecture Multi-Agents pour un Tuteur Intelligent*-Thèse de doctorat de l'Université de Paris 6- Spécialité Informatique, décembre 1990.
11. Glikman, V. (2000). A propos de la dimension pédagogique : les formateurs leurs rôles et leur fonction de médiation dans la FOAD, *Forum Formation à Distance et Technologies de la Communication*, pp. 42-48, 2000.
12. Hong, K.S., Lai, K.W., Holton, D. (2001). Web based learning environments: Observations from a Web based course in Malaysian context, *Australian Journal of Educational Technology*, Vol.17, N°3, pp.223-243, 2001.
13. Jennings, N.R., Sycara, K., Wooldridge, K. (1998). A Roadmap of Agent Research and Development, *Autonomous Agents and Multi-Agent Systems*, Vol.1, Boston, pp. 7-38, 1998.
14. Maes, P. (1997). Intelligent Software, *Proceedings of Intelligent User Interfaces*, Orlando, Florida, 1997.
15. McCalla, G.J. (1992). The search of Adaptability, Flexibility, and Individualisation : Approches to Curriculum. In Jones, Winne (eds.), *Adaptive Learning Environments.*, vol. F85, Berlin.
16. Modro, N.R., Paas, L.C., Rodriguez, A.M. (2002). The Brazilian Virtualizing Project: Using Virtual education to bring public school teachers closer to modern reality, *USDLA Journal*, Vol.16, N°12.
17. Murray, W.R. (1990). A Blackboard-based Dynamic Instructional Planner. *AAAI'89*, pp.434-441.
18. Ohlsson S. (1986). Some principles of intelligent tutoring, *Instructional Science*, Vol 14, 293-326.
19. Rasseneur, D., Jacoboni, P., Tchounikine, P. (2003). Visualisation multi points de vue d'une FOAD, Environnements Informatiques pour l'Apprentissage Humain, Strasbourg, France, pp. 559-562.
20. Sbihi, B (2003). The generation of code multi-target for the VBOOM method : Approche by filtering, The 2003 International Conference on Software Engineering Research and Practice (SERP'03), Las Vegas, USA 2003.
21. B.Sbihi, The integration of the points of view notion in UML, *AMO - Advanced Modeling and Optimization*, Volume 6, Number 1, pp 75-86, 2004.
22. Singh, P. (1994). *Multiagent Systems : A theoretical Framework for Intentions, Know-Hows, and Communications*. *Lecture Notes in Artificial Intelligence*, Springer-Verlag Publishers, Germany.
23. Tchounikine, P. (2002). Quelques éléments sur la conception et l'ingénierie des EIAH, 2èmes Assises Nationales du GDR I3, pp.233-245.
24. Vygotsky, L.S. (1978). *Mind in society: The development of higher psychological processes*. Cambridge, MA:Harvard University Press.
25. Wenger, E. (1987). *Artificial Intelligence and tutoring systems*. Los Altos, CA: Morgan Kaufmann.

On-line References

- CLAROLINE, (2002). Claroline, <http://www.claroline.net/>, 2002.
GANESHA (2002). Ganesha, <http://www.anemalab.org/ganesha/>,
2002. WEBCT (2002). WebCT, <http://www.webct.com/>, 2002.

Article received: 2005-05-13