

# CURRICULUM DESIGN ADAPTATION, EXECUTION AND MONITORING IN MOODLE

Antonio Garrido<sup>1</sup>, Lluvia Morales<sup>2</sup>, Ivan Serina<sup>3</sup>

<sup>1</sup> *Universitat Politècnica de València (SPAIN)*

<sup>2</sup> *University of Granada (SPAIN)*

<sup>3</sup> *Free University of Bozen-Bolzano (ITALY)*

*agarridot@dsic.upv.es, lluviamorales@decsai.ugr.es, ivan.serina@unibz.it*

## Abstract

Nowadays Internet is full of shareable learning objects that can be (re)used for curriculum design and composition of e-learning courses. But the ultimate goal is not to consider these objects in an isolated way, but to assemble them to get new synergies when designing new courses. In this paper, we show that AI planning technology offers very good capabilities for curriculum adaptation and can be combined with current learning management systems, such as Moodle. We introduce *myPTutor*, an architecture that integrates plan adaptation techniques with execution, monitoring and plan validation within Moodle. We aim at providing an automated intuitive mechanism for both teachers and students to manage course designs and enhance plan adaptation when flaws are detected during execution.

Keywords: e-learning, curriculum adaptation, personalization of learning routes, Moodle.

## 1 INTRODUCTION AND MOTIVATION

Learning Objects are all around. Google currently returns more than 20 millions of hits and hundreds of learning object (LO) repositories, such as <http://www.merlot.org>, <http://www.ariadne-eu.org> or <http://www.ocwconsortium.org>, which aim for full knowledge sharing and reuse. Moreover, the Web provides huge volumes of interoperable digital resources, implemented in different XML standards such as SCORM, LOM or IMS [1], [2], which are potential reusable LOs for e-learning, thus representing terabytes of knowledge. This huge development of flexible, reusable LOs and their availability on Web repositories has become a cornerstone to promote advances in many educational fields and disciplines. But, "*as individual words cannot independently produce meaning, LOs in themselves are insufficient to fully accommodate the different learning and studying styles, strategies and preferences of the students*" [3]. Therefore, offering personalized curriculum designs and learning routes to individual students' needs, profiles and preferences is essential to support better initiatives and experiences in e-learning. However, there is not a straightforward approach to manage LOs from heterogeneous sources because of differences in the ontology they use, the way they interrelate or how they work together.

There are different research approaches for course and curriculum composition, such as adaptive and dynamic courseware generation. Intuitively, the former selects LOs from a given repository to ensure that a student completes all the activities that a teacher deems important; this makes the overall objective instructor-centered [4]. The latter observes the student's progress during his/her interaction with a general course, and dynamically adapts it according to the student's requirements. The overall objective here is to assist students in navigating a complex information space in order to achieve whatever goal they choose, making this type of hypermedia technique student-centered [5], [6]. In other words, the adaptive generation focuses on selecting LOs from a given repository in a way which is appropriate for the targeted individuals, whereas the dynamic generation focuses on a more accurate browsing associated with an on-line course in an optimal order, where the optimization criterion may take into consideration the student's background and performance. As can be seen, these two approaches involve a separate vision: first to generate a "kind of tailored" course and second to execute, monitor and fully adapt such a course to the student's needs.

In this paper we face these two views in a joint way from an Artificial Intelligence planning perspective. Planning techniques offer very appealing possibilities for the development of e-learning environments that bring the right content to the right person, but also at the right time and with the right resources, which is usually missing in traditional e-learning. Hence, the advantage of using intelligent planning techniques is twofold. Firstly, they bridge the gap between the purely e-learning necessities and its

personalization, by making use of a flexible mapping of standard LOs and courses into standard planning domains, which are then solved by state-of-the-art planners and/or Case-Based Planning (CPB) methods [7]. Secondly, they go beyond the traditional e-learning insights and give support not only to adaptation and LO sequencing, but also to time-resource constraints and multi-criteria optimization metrics, which are important in realistic environments [8]. This raises a challenge for a successful integration with Learning Management Systems (LMSs) that facilitate the dynamic navigation of contents/LOs, monitor the students' progress when following their proposed learning routes, check whether some discrepancies (differences *w.r.t.* the expected state) appear and react to them to adapt the routes to the new necessities -by means of an adaptation process that does not ignore the original student's interests and tries to reuse the original route as much as possible [9]. In order to use this approach, LMSs need to be extended to provide capabilities to monitor and check the students' progress when executing learning routes. And it is not only a matter of plugging in a planner to find a route, but also to offer a mixed-initiative mechanism for both teachers and students to manage courses and automated tools to find out discrepancies in order to adapt the route. This is the underlying idea of the paper, which in essence contributes with:

- A knowledge engineering representation mechanism to map LOs, their LOM-based relationships and learning courses into a planning domain+problem description, which can be used by standard PDDL planners.
- A CBP approach for planning learning routes that does not require computing a new route from scratch, as it simply allows us to adapt high quality learning routes already executed in the past and stored in a database (plans library).
- A flexible way to make curriculum authoring easier because teachers can easily retrieve, adapt, improve and reuse courses previously generated.
- A simple translation of the resulting sequence of LOs into another standard representation which provides a usable manifest for on-line learning platforms, thus closing the e-learning cycle.
- A procedure in Moodle that facilitates the integration of our approach, named *myPTutor* (<http://servergrps.dsic.upv.es/myptutor>) to facilitate the dynamic navigation of contents. This also provides the foundations to: i) monitor the students' progress when following their proposed learning routes, ii) check whether some discrepancies appear, and iii) react to them to adapt the routes to the new necessities. Although we are still working on improving the functionality of the two first points, the third one makes use of the same CBP approach for plan adaptation and requires no additional changes.

The paper is structured as follows. In section 2 we introduce planning and how it can be used in an e-learning setting. Section 3 presents our entire approach in which we detail the structure and main elements, such as the translation from the e-learning model into the planning model and the planning module. Section 4 describes the integration of our approach within Moodle; this means to explain the new requirements that are necessary, how to deal with the execution and monitoring phases, and also to provide an application example. The paper concludes with section 5 in which we highlight the main challenges to be solved for integrating planning and LMSs and how we have faced them in our approach.

## 2 USING AI-PLANNING IN E-LEARNING

The main objective of AI planning is to come up with a proper plan, i.e. a fully or partially ordered sequence of actions, whose execution will transform an initial state into a final state in which all problem goals are satisfied, while possibly optimizing a (multi-criteria) metric. In an e-learning setting, the planning objective is to provide a student-centered solution (i.e. personalized learning to offer a specific learning route) by returning a flexible learning process where courses and LOs are tailored to the specific needs, learning styles, objectives, background and, in general, the profile of each student [6],[8],[9],[10],[11].

The planning formal representation allows us to precisely determine the evolution of the e-learning process (execution) and identify, in advance, any discrepancies between the expected and observed states of the world when the plan does not produce the expected result, or as a change in the goals to be achieved by the plan (monitoring). This monitoring phase examines the students' e-learning evolution and the changes related to external events for all the facts that describe their learning

process. A discrepancy with respect to the planned values could simply invalidate some temporal constraints, which can be simply repaired by a rescheduling of the plan, or can determine one or more failures in the plan which could also induce, in the worst case, a complete revision of the entire learning process. When a failure is discovered, the system may react by looking for a repair or aborting the current plan.

In order to overcome the limitations of planning from scratch and avoid a complete revision of the learning process, our plan revision phase adopts plan adaptation techniques to reuse and modify, if possible, the current execution plan so as to solve the new planning problem [7], [12]. Intuitively, adaptation consists in reusing parts of the retrieved plans to complete a new one that is similar to the original one. This allows teachers to easily check/validate the proposed learning route since they can simply consider and analyze the parts of the learning route that differ from the elements stored in a case base and that have been introduced in order to satisfy the new scenario. For example, if new LOs' prerequisites and/or users' goals appear, which do not fit in the initial students' requirements, adaptation could be worthier than generating a whole learning route from scratch both from a computational and teacher's effort point of view.

Note that different criteria can guide the definition of an e-learning route. As we will discuss below, in our current version we do not only try to find good quality plans that best fit the students' requirements, but also to minimize the number of LOs that have been introduced or removed *w.r.t.* the case base elements. Further, the relative importance of plan quality *w.r.t.* plan *stability* can be chosen by the teacher when the planner is executed. In fact, when replacing the plan an important consideration is *plan stability*. As proposed in [12], we use this term to refer to a measure of the difference a process induces between an original (source) plan and a new (target) plan. In the e-learning context it is extremely important to preserve as much as possible the LOs planned for each single student. In fact, it could be extremely disappointing if a completely new sequence of LOs is proposed to the students given a change in the current state. This should be avoided as much as possible, especially if the original LOs can still be reused.

### **3 PLANNING E-LEARNING ROUTES. OUR MIXED-INITIATIVE APPROACH**

#### **3.1 Structure and Behaviour**

The overall structure of *myPTutor* is depicted in Fig. 1. There are three roles. First, the course designer, who defines the course from scratch or by reusing LOs from standard repositories (typically known as SCORMs). Second, the students, who take part in a course and provide their background, preferences and learning necessities; this type of information can be requested every time or stored in an e-portfolio for repeated uses. Third, the teacher, who validates the plan provided by the automated planner. Note that this last role is essential in a mixed-initiative approach to make the interaction "automated vs. manual planning" possible. Clearly, creating an individual plan for every student requires a lot of effort for a teacher, but validating the plans (by a human teacher) before they are provided to the students also proves very advisable. Therefore, this mixed approach of automated plan generation and human validation offers nice advantages and reduces the associated burden.

On the other hand, there are three main automated processes. First, the compilation of the LOs of the course and the students' information as a planning model. This involves a knowledge engineering task described in section 3.2. Second, a planning technique that creates the personalized plans from scratch, or by adapting already existing plans in a plan library, which is described in section 3.3. Third, the LMS in which the entire approach is integrated. As explained in section 4, we use Moodle for this. Intuitively, Moodle provides the instructional design (plan) per student, thus allowing easy navigation and execution processes. An additional objective is to automatically monitor the student's progress and detect significant discrepancies between the current situation and the scheduled (expected) one [9]. In the event of a discrepancy, the planner tries to adapt the existing plan to fulfil the requirements in the new situation, then the new plan is shown again in the LMS and, finally, the cycle continues.

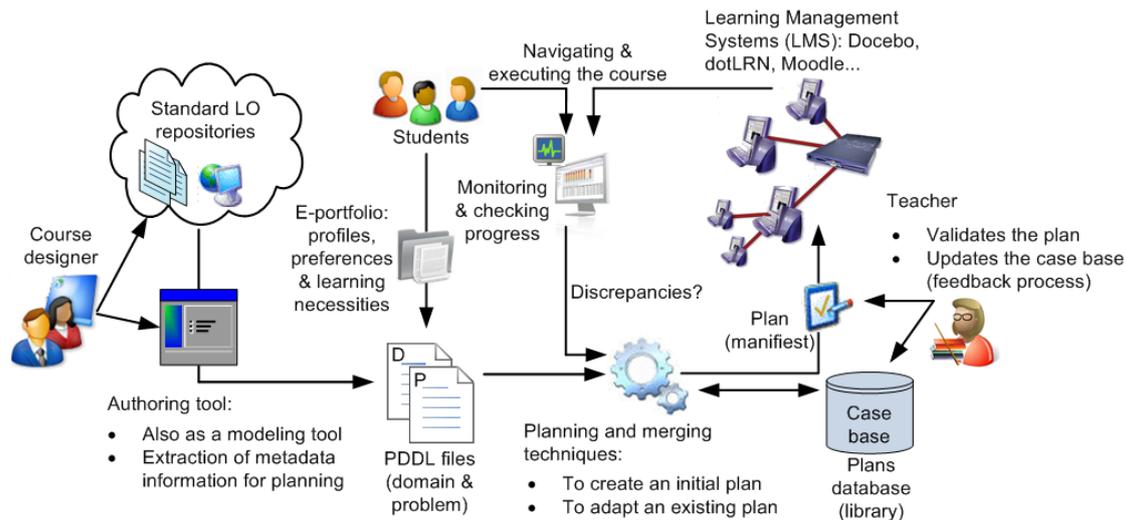


Fig. 1. Main schema of *myPTutor*.

### 3.2 Knowledge Engineering Mapping

From a knowledge engineering perspective, LOs can be considered as very simplified rules that involve some integrated knowledge provided by an expert practitioner. Loosely speaking, the *left side* of a rule has to be satisfied before the rule is applied, whereas the *right side* describes the results (in terms of what changes) after the rule application. For instance, if we design a LO for a “while loop in a programming language course”, we need to define: i) some previous knowledge on tests and repetition (i.e. preconditions or left side), ii) the learning content in itself (i.e. the material as a textual description, exercise, lesson, example or animation), and iii) the learning outcome (i.e. the fact that the student knows how to implement this type of loop, namely effects or right side). Consequently, repositories of LOs represent large amounts of knowledge, in the form of metadata, which emulate simple rules, whereas planning technology supports the reasoning mechanism to provide solutions, i.e. learning designs or routes. But before proceeding with this reasoning mechanism, an acquisition stage becomes necessary to compile the planning model. And thanks to this rule-style structure, we can establish a high similarity between LOs and PDDL (Planning Domain Definition Language) actions, thus allowing us to define a straightforward mapping to automatically build planning models.

A complete description of the LO metadata mapping to planning actions, that is the planning domain definition, is out of scope here (we refer the interested reader to [10] for further details). In short, metadata specification for LOs is usually described in a standard format, such as LOM [2], and comprises many items as depicted in Fig. 2. Despite this high number of items, just a few of them are essential for planning: i) the technical requirements (4.4 and 4.6), which represent the particular resources for the LOs; ii) the educational information about the student's learning type/profile, difficulty and duration (5.2, 5.8 and 5.9 respectively); and iii) the relations (7), which represent the content dependencies among LOs. In essence, these items represent the preconditions of a planning action. Hence, there is a simple metaphor: one LO is encapsulated by one action, and vice versa.

The collection of actions is generated by looping all over the LOs of the repository in a very efficient way, that is polynomial time. It is important to note that planning actions only need to be regenerated when their corresponding LOs change and remain unchanged when other unrelated LOs appear/disappear, which means an inexpensive knowledge engineering phase.

The planning problem definition is an analogous knowledge extraction process that acquires the relevant student's characteristics from his/her e-portfolio, i.e. the student's background (initial planning state), and learning outcomes (planning goals). Additionally, it is also possible to define a criterion to be optimized in the learning route, e.g. the shortest route or the one with the minimum number of LOs.

This automated compilation of a standard PDDL planning model (domain+problem) facilitates the use of independent solvers, which provides a nice approach to abstract the e-learning specific features from the planning details.

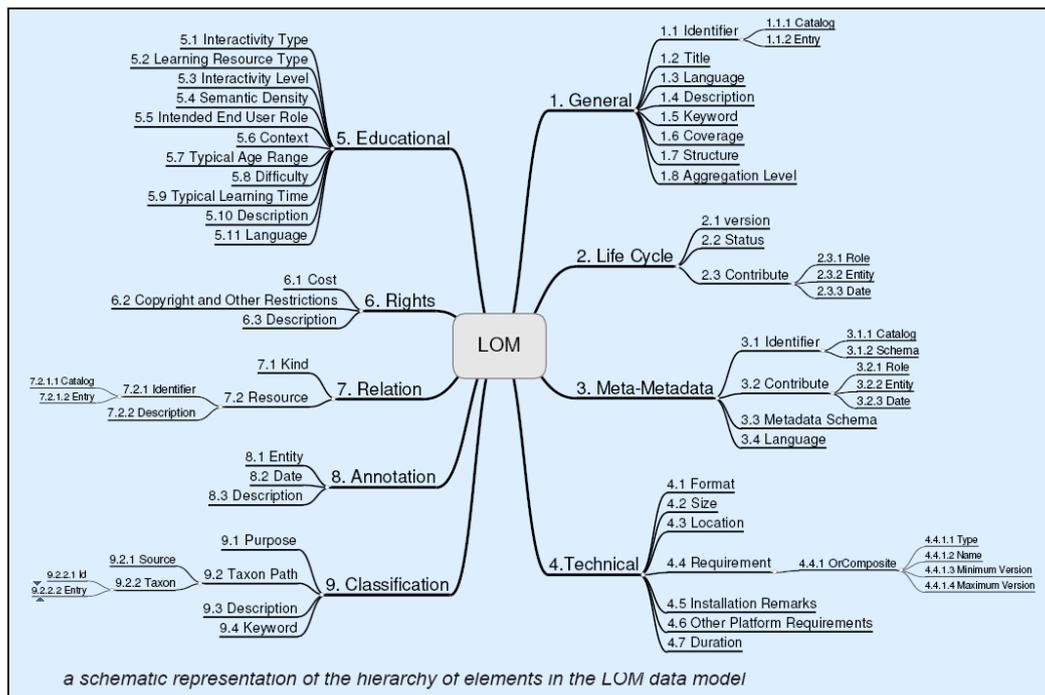


Fig. 2. Structure of the LOM conceptual data schema as defined by IMS [1], [2].

### 3.3 The Planning Tool

As regards to the planning tool for generating the initial learning route, *myPTutor* uses a CPB method [7] for the definition, memorization, retrieval and adaptation of learning routes. However, defining a high quality learning route is difficult; it depends on many variables, involving the LOs, the available resources, the students, and their preferences/learning styles.

Our objective is to help teachers, by means of a mixed-initiative approach, choose the most suitable learning route and semi-automatically adapt it in accordance with the students' goals and their individual features [10]. Case-based plan merging techniques are used in our approach to store learning routes (plans) previously validated by the teachers in the case base, as shown in Fig. 1. These solution plans can be manually generated by the teachers or simply generated by a domain independent planner such as LPG, SGPlan or by our case-based planner itself, and then validated by the teachers.

Similarly to other Case-Based Reasoning (CBR) systems, CBP is based on two assumptions on the nature of the world. Firstly, the world is regular: similar problems have similar solutions; as a consequence, solutions for similar problems are a useful starting point for new problem-solving. Secondly, the types of problems an agent encounters tend to recur; hence future problems are likely to be similar to current problems. Moreover, in order to reuse as many as possible parts of previously executed plans, we use *plan merging techniques*, which are based on the well-known *divide and conquer strategy*. In order to apply this strategy, we decompose the solution plans into subparts that allow us to satisfy every single goal and we store them in the case base as a new problem instance (if not already present). When a new e-learning planning problem must be solved, we search in the case base if a plan exists that already solves all goals. If such a plan does not exist we apply plan merging techniques that progressively identify (sub-)plans in the case base that can satisfy the goals. We progressively identify the unsatisfied goals and the corresponding (sub-)cases that allow to satisfy them, giving the preference to the (sub-)plans that allow us to improve the plan metric and that have been successful in a greater number of times in analogous situations.

## 4 INTEGRATION WITH MOODLE

Moodle, originally Module Object-Oriented Dynamic Learning Environment, is a Learning Management System, implemented as a free, open-source PHP Web application, to deliver and conduct online learning in a variety of settings -see <http://moodle.org> for more details. Moodle does not only provide a set of activity modules to build collaborative communities of learning around their subject matter, but

also an intuitive way to deliver content to students, such as standard SCORM packages, and assess learning using assignments or quizzes.

Moodle supports a long list of databases, but in practice, most systems are deployed using MySQL. The Moodle database is organized into over 200 related tables that we need to manipulate (and extend) to support curriculum design adaptation, execution and monitoring. Analogously, the Moodle code is organized into modules which implement different functionality such as a quiz, forum, blog, etc. Therefore, it is necessary to create new modules to display the planning functionality and coordinate its inputs/outputs with other modules, plus offering graphic interfaces for teachers and students.

In this section we present the changes that are required to support our approach and then include an application example to show how it works.

#### **4.1 Extending Moodle for Curriculum Design Adaptation. New Requirements**

The integration of our approach in Moodle requires changes in all the tiers of a typical three-tier architecture. First, data management requires five new tables to be defined in the database: i) the *planning precondition*, which can represent either a LO or an external precondition (e.g. resource or language requirement); ii) the *relation between LO and precondition* to store which are the necessary preconditions to satisfy a LO; iii) the *question* to be asked to the student to find out if (s)he fulfils a precondition (e.g. which is the initial level of English of a given student); iv) the *planning goal*, which represents the learning objective (e.g. a competence level that a teacher defines); and v) the *relation between the goals and the course* (e.g. a teacher deems *goal1* essential for a course but *goal2* optional).

Second, the application processing tier is responsible for creating an interface between database information for being capable of accessing and modifying the quiz module information according to preconditions' requirements, as well as modify and monitor the SCORM module contents and activities. Another interface is created between the planning techniques and the planning module; these techniques are accessed as a Web service that decides whether a planning, replanning or plan repair technique must be used. Furthermore, a set of services for translation are necessary: i) SCORM information into PDDL, ii) planning outputs into calendar formats that allow their validation, and iii) an integration of these validated plans into the Moodle data representation for each student.

The third tier is the presentation of the planning module (GUI) which consists of two pages that are shown according to the role that the user has in the current course, either a student or a teacher. For teachers, the page allows selecting the main objectives of the course represented as individual SCORM objects, and the particular compulsory goals the student must reach in order to achieve each objective. These goals have particular preconditions that the teacher has to define together with some questions to find out the students' background. Once the teacher has prepared this information and all the registered students have answered the questions, the teacher will have the opportunity to automatically plan an activity schedule for these students, download the solutions (plans) in different calendar formats and monitor the students' activity progress in order to know when setbacks happen during the execution of these plans. On the other hand, when students use their planning page they have the opportunity to select the optional goals they prefer, access the mandatory quiz, replan their activities schedules according to their particular preferences and access the SCORM objects affected by the planning process in order to continue with the execution of the assigned activities.

#### **4.2 Execution and Monitoring**

Moodle provides an intuitive navigation GUI throughout the student's instructional design, thus making the execution process a simple issue. Monitoring the students' progress to check if there exist discrepancies is however more challenging. Discrepancies may appear due to changes in the students' profile information, the temporal constraints, the resource availability, or the execution of the LOs in themselves. For instance, during the course execution the student might fail a test, that is, a kind of *checkpoint* used to evaluate his/her competence level. If this level shows a low score, the student's performance will not be enough to continue the course or to attain the learning outcomes, and a replanning stage will be necessary to adapt the learning route from the new state (by adding or deleting LOs). Note that the changes in students' background, learning styles and temporal constraints must be modified directly by the students using Moodle's forms and tools. On the other hand, changes related to the resource availability and scores of evaluation activities are updated by the teachers.

Basically, a discrepancy means that the real situation is not as expected and the remaining part of the plan may become invalid [9]. We use *VAL*, an automatic validation tool for PDDL plans downloaded from <http://planning.cis.strath.ac.uk/VAL>, which finds out if an action cannot be executed in a given state and why, i.e. which preconditions are not satisfied. At this moment, the teacher can fix the plan by hand or in an automated way. In the last case, namely automatic adaptation, a new planning problem is created: the initial state represents the current state, whereas the problem goals can remain the same or modified by either the teacher or student. This new problem can be solved by our planning tool introduced in section 3.3 or by other domain independent planners. Once a new plan is computed, it must be validated by the teacher, stored in the plans database (see Fig. 1) and sent back to Moodle's navigation GUI.

### 4.3 Mixing Altogether. An Application Example

In this section we show how to use our approach in Moodle as a short tour (see Fig. 3, which considers a Natural Science course). It all starts with the definition of the course by the teacher, as shown in Fig. 4. There are basically two pages for doing this. In Fig. 4-1 the teacher defines the contents (SCORMs) of the course. The goals of these SCORMs are automatically recovered from the database and (s)he decides which the compulsory goals for this course are; in our case "genetics" and "structure" are essential and the others are optional. Attaining these compulsory goals requires some initial preconditions on the students which are also shown in that page. Now the teacher creates questions (or reuse them from the question bank of the database) to be asked to the students to find out whether they satisfy the required preconditions; note that the answers can be taken from the student's portfolio or forced to be evaluated time and again. In our example of Fig. 4-2 there are questions to know the performance level of the student, the Spanish level and the multimedia requirements. After doing this, the course will be created by the planner and the teacher can get an initial plan for the students enrolled on the course. Afterwards, the plan can be downloaded in different formats.

The student proceeds in an even simpler way as shown in Fig. 5. If a student enrolls on a course, there will be compulsory goals (previously defined by the teacher) and optional goals that each student may freely choose (see Fig. 5-1). According to the set of goals chosen, the student needs to answer a quiz as the collection of questions that the teacher defined for the required preconditions. In our example, the student needs to give his/her initial value on performance and Spanish level, and his/her multimedia capabilities. After doing this, a fully-tailored plan is automatically generated by our planner; this plan can be directly shown to the student or first checked by the teacher. The navigation and execution of such a plan is typically depicted in Moodle as a scheme of activities (Fig. 5-2). If a discrepancy is found during the monitoring phase and some remaining activities become invalid, they are displayed in a strike font type. If this happens, a replanning/rescheduling stage becomes necessary to adapt the plan to the new scenario, as discussed above.



Fig. 3. Moodle's main page to choose among the activities.

Finally, it is important to note that both the teacher and students can see the plan using the Moodle GUI or download the plan in other formats. Currently, *myPTutor* offers this export functionality to PDDL solution files and XML Microsoft Project. In the particular case of Microsoft Project, this is very useful as the plan can be shown as a Gantt Diagram with all the precedence relations in a visual way.

For instance, Fig. 6 shows the plan for three students with the corresponding sequencing and calendar. This also allows the teacher to change some ordering relations according to his/her experience, provided they do not create an inconsistency, and export these changes back to the plan stored in Moodle. Since we have implemented PDDL to XML Microsoft Project translators and vice versa, the plan that is automatically generated by our planner can be graphically tuned by the teacher and then given back to the student.

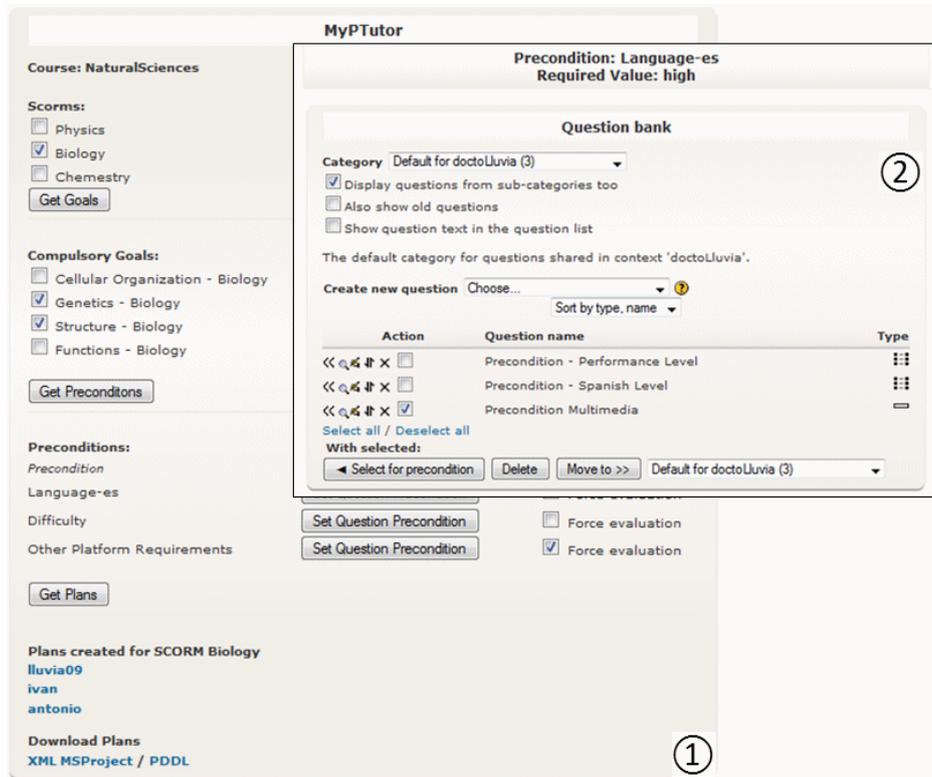


Fig. 4. *myPTutor* GUI for teachers: defining the course, SCORMs, goals and preconditions in Moodle.

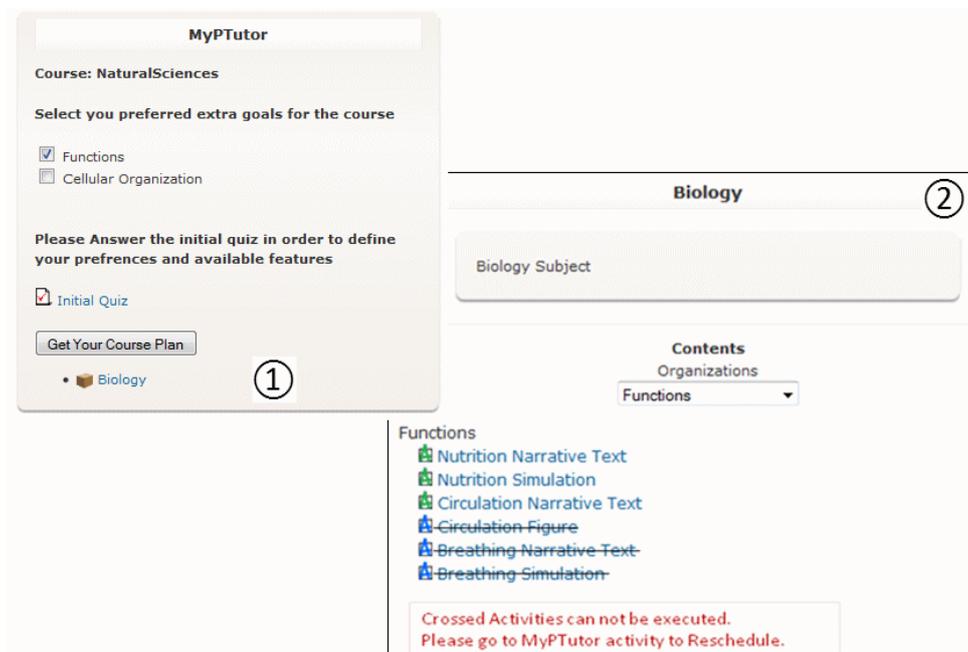


Fig. 5. *myPTutor* GUI for students: selecting optional goals and generation of the plan.

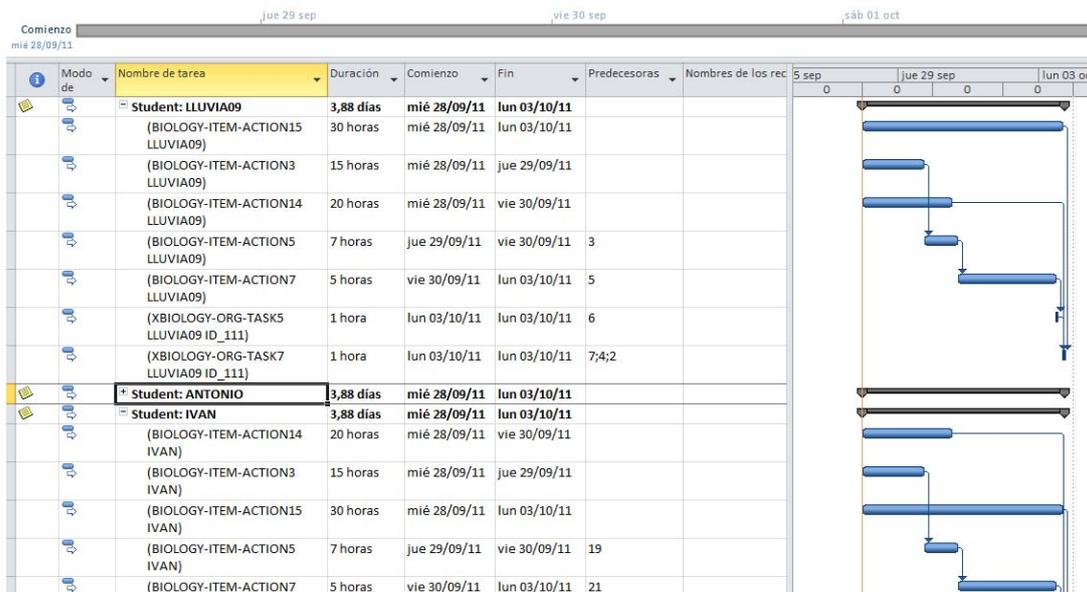


Fig. 6. *myPTutor* export option to Microsoft Project: plan as a Gantt diagram.

## 5 CONCLUSIONS

We have presented an approach that integrates the use of AI planning technology within Moodle to provide fully tailored e-learning routes that fit the students' needs and preferences. This raises important challenges to be solved:

- How to translate the e-learning standard into a planning model in a transparent way? Leaving apart most pedagogical aspects, we use an automated straightforward knowledge engineering mapping that transforms each LO (with its dependency relations) into a PDDL planning action (with its preconditions and effects).
- Which are the most appealing planning techniques? After generating the planning model, any type of planning-from-scratch algorithm could be used to generate learning routes. However, in an e-learning setting where preserving the original plan as much as possible is vital (plan stability), a CBP approach for plan adaptation demonstrates more adequate in all our tests [9]. Also, the use of planning techniques allows us to handle time+resource constraints and multi-criteria optimization metrics.
- How to perform a fully integration within a LMS? All LMSs offer very nice options for showing plans (i.e. instructional designs) in graphical or schematic ways. Hence, the execution of these plans is not particularly difficult. But, how to monitor such an execution and provide feedback when the state differs from the expected one is more difficult. We use plan validation techniques to analyze when some preconditions do not hold in a given state and, therefore, actions (LOs) cannot be executed anymore. In such a situation, we use our plan adaptation approach to fix the part of the plan that represents a flaw and try to reuse as many original LOs as possible. This is good for both the teacher (who has already validated the original plan and does not want to validate a brand new plan) and the student (who expects a plan without many significant changes).

We have implemented all these features in Moodle and demonstrate they couple well together. We are now working on designing a clean GUI and extensively validating the functionality of the approach. These points and a thorough testing stage are part of our short-term work.

## ACKNOWLEDGEMENTS

This paper was partially funded by the Consolider AT project CSD2007-0022 INGENIO 2010 of the Spanish Ministry of Science and Innovation, the MICINN projects TIN2008-06701-C03-01 and TIN2011-27652-C03-01, the Mexican National Council of Science and Technology, the Valencian Prometeo project 2008/051 and the BW5053 research project of the Free University of Bozen-Bolzano.

## REFERENCES

- [1] IMS. (2008). IMS Global Learning Consortium. Available at <http://www.imsglobal.org> (accessed Sept. 2011).
- [2] LOM. (2005). Draft Standard for Learning Object Metadata. IEEE. Available at [http://ltsc.ieee.org/wg12/files/IEEE\\_1484\\_12\\_03\\_d8\\_submitted.pdf](http://ltsc.ieee.org/wg12/files/IEEE_1484_12_03_d8_submitted.pdf) (accessed Sept. 2011).
- [3] Polsani, P.R. (2003). Use and Abuse of Reusable Learning Objects. *Journal of Digital Information* 3(4). Available at <http://journals.tdl.org/jodi/article/view/89/88> (accessed Sept. 2011).
- [4] Abdullah N.A., and Davis, H. (2003). Is Simple Sequencing Simple Adaptive Hypermedia? *Proc. ACM Conference on Hypertext and Hypermedia*, pp. 172-173.
- [5] Perez-Rodriguez, R., Rodriguez, M.C., Anido-Rifon, L.E., and Llamas-Nistal, M. (2010). Execution Model and Authoring Middleware Enabling Dynamic Adaptation in Educational Scenarios Scripted with PoEML. *Journal of Universal Computing Science* 16(19), pp. 2821-2840.
- [6] Ullrich, C., Lu, T., and Melis, E. (2009). Just-In-Time Adaptivity Through Dynamic Items. *Proc. User Modeling, Adaptation, and Personalization Seventeenth International Conference (UMAP)*. Springer, LNCS 5535, pp. 373-378.
- [7] Serina, I. (2010). Kernel Functions for Case-Based Planning. *Artificial Intelligence* 174, pp.1369-1406.
- [8] Garrido, A., and Onaindia, E. (2010). On the Application of Planning and Scheduling Techniques to E-learning. *Proc. Int. Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems (IEA-AIE)*. Springer, LNAI 6096, pp. 244-253.
- [9] Morales, L., Garrido, A., and Serina, I. (2011). Planning and Execution in a Personalised E-learning Setting. *Proc. Conference of the Spanish Association for Artificial Intelligence (CAEPIA)*. Springer, LNAI 7023, pp. 232-242.
- [10] Garrido, A., Morales, L. and Serina, I. (2011). Applying Case-Based Planning to Personalized E-learning. *Proc. Int. Workshop on Distance Education Technologies (DET)*. Knowledge Systems Institute Graduate School, pp. 228-233.
- [11] Brusilovsky, P., and Vassileva, J. (2003). Course Sequencing Techniques for Large-Scale Web-Based Education. *International Journal Continuing Engineering Education and Lifelong Learning* 13(1/2), pp. 75-94.
- [12] Fox, M., Gerevini, A., Long, D. and Serina, I. (2006). Plan Stability: Replanning versus Plan Repair. *Proc. 16th Int. Conference on Automated Planning and Scheduling (ICAPS-2006)*. AAAI Press, pp. 212-221.