Automated planning for personalised course composition

Antonio Garrido, Eva Onaindia and Oscar Sapena
Phone: (+34) 963877007   Fax: (+34) 963877359
Universidad Politécnica de Valencia
Camino de Vera s/n, 46022, Valencia (Spain)
{agarridot, onaindia, osapena}@dsic.upv.es

Abstract

Authoring tools for building Intelligent Educational Systems must provide support to ensure flexibility, adaptability of content to the user profile, reusability and sharing of learning objects. These facilities are essential to develop automated decision processes for providing course compositions tailored to the specific characteristics of each individual learner. We present a LOM-compliant learning approach that uses an automated planning process to create personalised learning courses while giving special attention to the development of reusable learning objects.

1. Introduction

Authoring tools for building Intelligent Educational & Tutoring Systems must provide support to ensure flexibility, adaptability of content to the user profile, reusability and sharing of e-learning objects (LOs) [5]. These are also key features in the development of automated decision processes for providing course compositions tailored to the specific characteristics, goals and preferences of each individual learner.

The selection and composition of LOs require meta-data that denote a level of semantic specification enough to enable consistent runtime automated semantics [6]. This is especially relevant in the case of automated planning processes where the access to specific content of a given course requires that all resources are accurately described by structural relationships which reflect the logical sequence of content [7]. However, since meta-data labelling is usually an arduous task and a not often attained goal, this has led to repositories where the learning content does not meet the necessary requirements to serve as the basis for common automated learning activities. In addition, the roles of relationships are not free of ambiguity, which seriously hampers the possibilities of consistent composition [6]. The efficiency of a planning process relies, among other things, on the accuracy of the LOs relationships. Therefore, a major issue in building a personalised course (plan) is how to provide LOs with appropriate meta-data annotations in order to increase the automation level for the composition of LOs and services (planning).

In this paper, we present a LOM-compliant learning approach [4] that uses an automated, adaptive planner to create personalised learning courses that are portable to any IMS-LD run-time environment while also focusing on the modelling of reusable LOs. Our motivation for reusability is to achieve a better identification of LOs when searching for sets of related resources, for which it is crucial to create a shared semantic basis for metadata elements usage. This semantic information can be expressed through structural relationships between LOs to facilitate knowledge sharing and reuse. Our approach follows the IMS specification and facilitates the completion and extension of meta-data records of LOs, specially those ones related to the structural and logical relationships of LOs. Particularly, we present a contribution to:

1. Import LOs from sharable repositories.
2. Improve meta-data labelling by extending the instructional design rationale, thus promoting a personalised access to the LOs.
3. Design our own LOs.
4. Export the improved content into sharable repositories.
5. Create personalised goal-oriented learning routes through an automated planner.

2. General schema

The general schema of our approach is depicted in Figure 1. First, a teacher uses our modelling tool to design, from scratch or by reusing LO collections, a pool of LOs. These LOs can be part of one or more courses, defined as instructional designs, based on an explicit, topic-centered structured representation of the learning domain. Note that this information is profile-independent, so it is valid for students with different learning styles. Second, students choose a whole course to follow, or simply the learning outcomes they are interested in from the course, indicating their personal characteristics (profile), background and preferences. The course structure, together with the students' information, form a planning problem, which is automatically generated and encoded in PDDL (Planning Domain Definition Language) format [2]. Finally, an intelligent planner takes the planning problem and generates a plan (learning route) per student, as a sequence of ordered activities which will lead each student to attain his/her learning goals.

As can be seen from above, the key elements of our approach are the modelling tool and the AI planner.

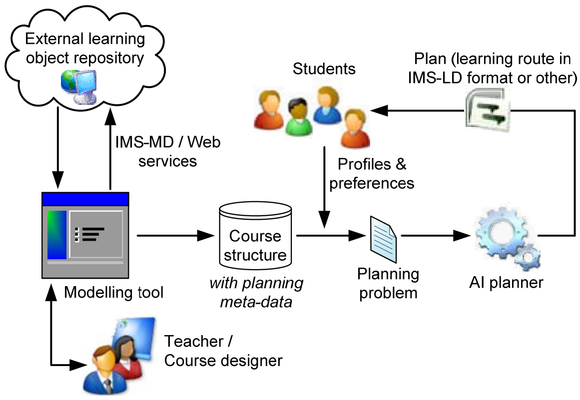

Figure 1. Schema of our approach at a glance

2.1. The modelling tool

Our modelling tool allows the designer to create e-learning courses by means of intuitive drag&drop graphic components and user-friendly input forms. In our planning perspective, we identify three main elements: tasks, concepts and learners' profiles. Basically, tasks encapsulate the LOs used to achieve the concepts or learning outcomes. Concepts act as prerequisites and/or learning outcomes of a task. However, as not every task is equally appropriate for all students, the designer may adapt the form and content of tasks to create profile-dependent tasks. For instance, a task may produce better or worse outcomes depending on the profile (learning style) of the student [1]. Consequently, the tool allows the instructor to explicitly model and label the structural relationships between tasks, concepts and the adaptation of tasks to the learner's typology and particularities. Furthermore, instructors can also specify the learning material (slides, handouts, etc.) required for the tasks, the type of this material (visual, multimedia presentation, etc.) and the resources, to achieve if not freely available, to use the material (computers, books, oscilloscopes, etc.).

In order to make the design stage easier, the tool defines three views according to the main component is being modelled:

- The **conceptual** view defines the outline of the course contents and their relationships, i.e. what knowledge is going to be learnt.
- The **task** view applies a decomposition of concepts into tasks, that is, the set of LOs the student needs to use. Thus, this view defines how the knowledge is going to be given.
- The **adaptation** view encompasses all the previous components, on top of which the instructor represents the information about the learners, their profiles and background, as well as the context characterisation, i.e. who the knowledge is targeted to and under which resources.

Meta-data for planning. An important advantage of our tool is that it can handle LOs encoded in the IMS-MD (learning resource meta-data specification) standard [3], thereby allowing the designer to reuse many of the LOs available on the Web. The standard IMD-MD provides several meta-data labelling specifications for LOs, some of which are essential for planning learning routes fully personalised and adapted to the students' preferences/necessities. Actually, the more accurate the meta-data labelling of the LOs is, the easier the adaptation to the student and quality of the course will be. Typically, the meta-data specifications required for planning are:

- Information about the student's learning style.
- Required resources, such as special equipment.
- Typical learning time, i.e. duration of the LO.
- Relationships among LOs and their types.

The first three items are self-descriptive. The relationships include hierarchical structures and ordering relations based on content dependencies. The hierarchical structures use the *IsPartOf* relationship, which represents a hierarchical aggregation of LOs. For instance, if 'LO2 *IsPartOf* LO1' and 'LO3 *IsPartOf* LO1' both LO2 and LO3 are necessary in a learning route that includes the (higher level) LO1. The basic ordering relations are *IsBasedOn* and *Requires*, which represent orderings and causal dependencies on previous knowledge. For instance, if 'LO5 *Requires* LO4' LO4 must be finished before starting LO5. Our flexible tool allows the designer: i) to download the LOs from the Web repositories; ii) to validate and enhance these and other meta-data specifications such as title, description, keywords, etc.; and iii) update the Web repositories with the improved LOs.

In particular, the tool offers three types of causal dependencies, *Requires*, *IsBasedOn* and *References* and all their inverse relationships too, as in LOM terminology [4]. We interpret the two first relations as follows: all the *Required* elements, and at least one of the *IsBasedOn*, have to be completed before initiating a LO. The designer might also recommend other previous LOs by means of the *References* relation. This relation does not denote a strong requirement but a recommendation (soft requirement) to complete a LO before proceeding with the next one.

2.2. The AI planner

Once the course and the students' information are modelled, a planning problem in PDDL is generated. It is important to highlight that no special expertise is required to develop the planning problem, as it is straightforward and automatically generated from the structures and meta-data defined in the modelling tool.

Due to space limitations we only give a brief description of the planner. We have implemented an ad-hoc planner that performs a heuristic backward search starting from the learning goals. It performs a multi-criteria optimisation *w.r.t.* the length of the learning route, its resource cost, or any combination of them. Therefore, the planner finds a sequence of learning activities, fully tailored to the students, which tries to optimise the user-specified criteria.

We use a classical goal-driven backward strategy that builds each learning route by regressively adding the profile-dependent LOs (tasks of the task view) that satisfy the goals. We apply a relaxed planning-graph-based heuristic as a guide mechanism, and it is here where the efficiency of our planner relies. A relaxed planning graph is a multi-level directed graph, where each level represents the value of the function to be optimised of the current partial solution. Hence, if the planner is optimising the length of the plan, each level represents the duration of the partial plan. The graph is called relaxed because some of the restrictions are ignored during its construction; tasks of the same student overlap in this graph due to the relaxation (obviously, a student cannot execute more than one task at a time). Consequently, the plan extracted from the relaxed planning graph is a lower bound of the optimal plan. When the plan is *stretched*, i.e. tasks of the same student are sequentially placed, it represents a very accurate solution. Actually, our planner behaves in an incremental (anytime) way; the more time it is allotted, the better the quality of the solution is. This process continues until the search space is exhausted (completeness preserving property), at which point we can guarantee the best solution has been found (property of optimality) or that there is not a feasible solution.

3. An application example

In this section we model part of the introductory Information Technology volume described in the ACM 2008 Computing Curricula (see *www.acm.org/education/curricula-recommendations*) from two perspectives, the teacher's and the student's. We present a short description, with the pros and cons, on how to use the modelling tool and the planner in a small laboratory example. Next, we present a preliminary evaluation of a real example.

3.1. From the teacher's perspective

Our example on the ACM introductory IT volume does not try to be exhaustive but simply illustrate how a teacher defines the three views, as represented in Figure 2. The conceptual view is simply used as a general conceptual map to help the teacher organise the contents of the course, which are fully specified in the task view with all the necessary LOs. In this example, some LOs were imported from LOM objects in the IMS-MD standard format, whereas others were directly created in the tool. As can be seen, the teacher can create new LOs or edit all the meta-data specifications of existing LOs. It is particularly important to fill in the relationships among LOs, which can be easily done through the graphic representation (see Figure 2).

The third view comprises a total personalisation of LOs, requirements and effects adapted to profiles. Personalisation means specifying which LOs are more suitable for which types of learners, the required and achieved competence level of concepts, and the resources required for each LO. In the adaptation view of Figure 2, the teacher exemplifies this by means of five learning outcomes or abilities: analysis of problems (*An.Problem*), understanding issues (*Und. Issues*), analysis of computing impact (*An.Impact*), professional development (*Prof.Develop*) and use & apply technical concepts (*Use&App*). The teacher also models three profile types: reasoning (inductive or deductive), organization skills (high, medium or low) and processing (visual or verbal), inspired on Felder's classification [1], or freely chosen by the instructor. For instance, LO 0 produces a competence level of +90% in the ability *An.Problem* if the student is inductive and +70% if the student is deductive. Similarly, LO 1 is appropriate for students with organisation skill={high, medium} but not low, and the level of the outcome *Und.Issues* depends on the value of such a skill. Additionally, LOs may have different duration and/or use different resources, which

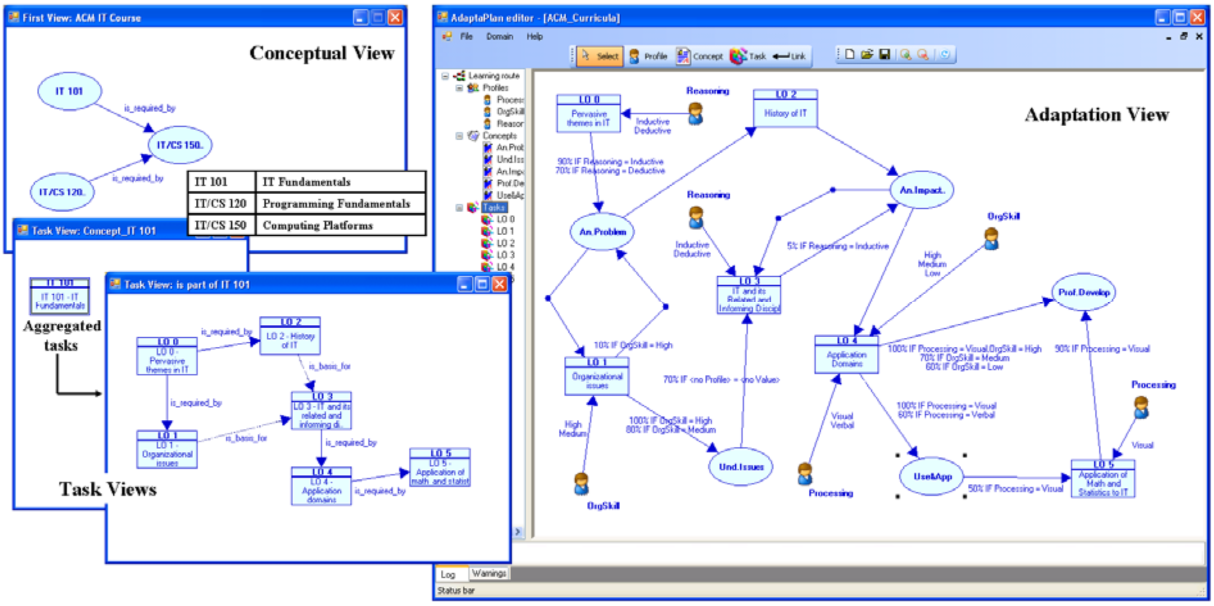

Figure 2. The three (conceptual, task and adaptation) views of our example

may impose constraints on the problem. For instance, if a LO requires a computer and a student has not his/her own computer, the academic centre will have to provide them one, which implies an utilisation cost the planner must account for.

Note that the decisions about the learning styles and their adaptation correspond entirely to the instructor. Pedagogically speaking, this entails a rethinking stage, which initially requires some effort in redesigning the LOs to specify their scope and establish which ones are more appropriate for each learning style. The tool shows here to be quite helpful in assisting the teacher in tasks such as labelling and assembling LOs from sharable repositories into a greater interoperable unit.

One problem here is that some teachers show reluctant to adapt this way of proceeding because of the required start-up effort. Clearly, at the beginning this approach requires a change of mind *w.r.t.* the conventional way of teaching. But, as long as students' roles are more dynamic, autonomous, and the self-learning is more demanded, this process of selection, construction and composition of coherent LOs becomes absolutely indispensable.

3.2. From the student's perspective

Unlike teachers, students do not need any previous training or change of mind. They only have to classify themselves in one or more learning styles, and define their background, preferences (in terms of the learning outcomes they want to achieve) and the resources they dispose of. For instance, let us assume two students interested in learning the IT Fundamentals (LO IT 101 of Figure 2). *Std1* is inductive, visual, with org.skill= high and a multimedia computer, and requires a minimal value of 50% in all the concepts (learning outcomes). *Std2* is deductive, verbal, org.skill= medium, with an initial 20% of competence level in *An.Problem* (background or previous knowledge), and whose learning goals are to attain 100% and 90% of *Use&App* and *Prof.Develop*, respectively. These high competence levels means *Std2* is interested in obtaining a high grade in the outcomes.

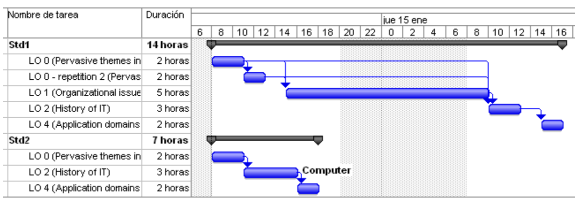

Figure 3. Learning route for the two students

The planner computes a plan for each student as a collection of learning activities conducted within an environment consisting of the LOs that make up the student's learning route, the relationships among LOs an the required resources. We have currently opted for a straightforward alternative to visualise this result: export the plan to the XML format of Microsoft Project. This is a well-known application to all students, and provides great flexibility to manage learning routes as Gantt diagrams and trace the project, similarly to the student' interface of intelligent tutoring systems. The learning routes for the two students of this example, with five and three LOs respectively, are shown in Figure 3. Once the students have the learning route at their disposal, their only mission is to execute the plan and complete the activities as suggested.

3.3. Results of our ongoing experiment

We are currently testing our approach in a 10-hour complementary AI course for eight students before attending an ‘AI planning’ PhD course. The students come from different universities and countries, so they have different background on AI topics. The idea is to level out their knowledge on AI before starting the PhD course. Additionally, students may have particular interests in going deeper on some specific topics.

The eight students used our tool to obtain a personalised learning route. After introducing their personal data, characteristics, background level and particular interests, they got a plan consisting of several LOs tailored to their lacks, needs and personal preferences. This way, some of the plans were more search-oriented, while others more KBS-oriented.

The success of our tool cannot be directly assessed through the students' grades as following a learning route does not necessarily mean a better score. Actually, better grades can be obtained by requiring higher values in the competence level of the learning outcomes, which forces the planner to build routes including reinforcement LOs. Consequently, it is not easy to measure the goodness of our tool via the students’ records. Yet, the main feedback is the students’ satisfaction when working with LOs that address their preferences and subjects of interest. Loosely speaking, the selected LOs in the personalised plan fit their learning styles and let them work at their own pace, thereby improving their personal instruction. We intend to use this feedback for validating the adaptability of the learning route and adjust the competence levels to the students’ profiles.

4. Conclusions

Our approach elaborates in gathering and composing a collection of LOs tightly interrelated and interoperable by combining: i) a modelling tool to reuse, incrementally design and label the LOs, and ii) an intelligent planner to accommodate the LOs to different learning styles and resources. These are two functionalities that most educational systems lack.

Our approach shows beneficial for both teachers and students. In the medium/long term, it is worth the teachers’ effort in designing the course so as to have reusable LOs that can be further used for planning other routes in distance e-learning environments. In the short term, students exploit the advantages of personalised courses, thus increasing their motivation and promoting more engaging learning experiences.

Our future work is to apply this approach in more courses and incorporate a module for monitoring the execution of the students’ learning routes, similarly to intelligent tutoring systems. This module will allow teachers: i) to get more experience for adjusting the competence levels to students’ profiles, and ii) to check whether everything happens as expected or, otherwise, an incident has occurred, e.g. a student takes longer than expected to complete a LO, a student does not achieve the expected competence level, etc. If such an incident invalidates the learning route, the planner must compute a new learning route (replanning mechanism) to let the student proceed and achieve the goals from the new situation.

Acknowledgements

This work has been supported by the Spanish projects MEC TIN2005-08945-C06-06, MICINN TIN2008-06701-C03-01 and Consolider-Ingenio 2010 CSD2007-00022, and by the UPV PAID-05-08.

References

[1] R.M. Felder, and L.K. Silverman, “Learning and Teaching Styles in Engineering Education”, *Engr. Education*, Vol 78(7), pp. 674-681, 1988.

[2] A. Gerevini, and D. Long, “Plan Constraints and Preferences in PDDL3”, *Proc. ICAPS - International Planning Competition*, pp. 7-13, 2006.

[3] IMS Global Learning Consortium, available at *www.imsglobal.org*, (accessed January 2009).

[4] LOM, “Draft Standard for Learning Object Meta-data”, IEEE, Available at *ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf*, (accessed January 2009).

[5] T. Murray, “Authoring Intelligent Tutoring Systems: An analysis of the state of the art”, *Int. Journal of Artificial Intelligence in Education,* Vol. 10, pp. 98-129, 1999.

[6] S. Sánchez-Alonso et al., “Design by Contract-Based Selection and Composition of Learning Objects”, *Computers and Education: E-learning, From Theory to Practice*, 179–191, 2007.

[7] J. Brase, and M. Painter, “Inferring Meta-data for a Semantic Web Peer-to-Peer Environment”, *Educational Technology & Society, 7(2), 61-67, 2004.*