Generation of exercises within the PERLEA project

Marie Lefevre, Stephanie Jean-Daubias, Nathalie Guin
Université de Lyon, France
Université Lyon 1 – LIRIS, CNRS, UMR5205
43 bd du 11 novembre 1918, 69622 Villeurbanne Cedex, France
{Marie.Lefevre, Stephanie.Jean-Daubias, Nathalie.Guin}@liris.univ-lyon1.fr

Abstract

The research we have carried out relates to the personalization of learning thanks to the exploitation of learners' profiles through the PERLEA project. We were aiming at designing a module managing the proposition of paper and pencil activities. For this purpose, we suggested a typology of exercises that can be given to a learner, as well as the architecture of generators allowing the creation of all of these exercises.

1. Introduction

One of the issues at stake in the research on the Interactive Learning Environments (ILE) is the personalization of learning. This personalization uses in particular learners' profiles assembling information about the learners, thus allowing us to characterize their knowledge, skills, perceptions and/or their behaviors. These data are collected or deduced from one or several pedagogical activities, computerized or not [6].

The personalization of learning, whether in the context of classic teaching or ILE, can relate to the interactions between the teacher / the environment and the learner as well as pedagogical activities assigned to the learner. In our research as presented here, we will focus on the second point.

In order to personalize pedagogical activities offered to the learner using a learner's profile, we can either use knowledge-based systems to generate the pedagogical activities best-suited to the profile, or provide the teachers with tools enabling them to perform this task themselves. Our wish is to link these two options in Adapte, a module of the PERLEA project.

After the presentation of a scenario of utilization, we will present the PERLEA project and its software environment. We will then focus on the Adapte module. To design this module, we have suggested a typology of exercises that can be given to a learner, together with the architecture of eight generators allowing the creation of all of these exercises. We will detail these two points before moving on to their implementation and to our validation.

1.1. Scenario of utilization

A teacher uses in his classroom an ILE on geography with his eight-year-old pupils. At the end of the learning session, this ILE generates a profile for each learner. In addition, the teacher organized for all his students the national assessments due in the beginning of year. These assessments have generated a diagnostic on the achievements, mistakes and difficulties of each pupil in mathematics and French.

Thus, the teacher has for each pupil several profiles from different sources, ILE and pencil and paper, and for several disciplines. He or she wants to use these profiles globally so as to provide, for each pupil, personalized exercise sheets. These sheets enable learners to be self-reliant when working and this in several disciplines.

Currently teachers cannot follow this pattern easily. Actually, there is no existing tool that would enable teachers to use data from ILE externalizing their learners' profiles, or to link this data to the pencil and paper profiles they themselves managed. Moreover, teachers can produce personalized exercise sheets but they then have to either create exercises themselves, or take existing exercises and manually adjust them to their needs and their working methods. They also have to decide for each student which exercises to put in his or her sheets. This work requires a great involvement from the teacher.

1.2. The PERLEA project

The PERLEA project aims at improving the integration of ILEs in education by providing links between the use of ILEs and teachers' everyday practices. In order to do so, we are interested, in a generic way, in the profiles of learners and their use a posteriori for the management of learners and the personalization of learning [6].

Hence we want to develop an environment that would enable teachers to manipulate existing profiles. This environment consists of two phases: the integration of existing profiles and the management of thus restructured profiles.

In order to explain this mechanism, let us take the example of our teacher with profiles from the ILE on geography and profiles from national assessments of mathematics and French. Reusing profiles requires knowing their structure. The teacher then defines a unique frame of profiles describing the information contained in the two types of profiles. This profile frame contains information on the three disciplines. Next, the teacher specifies to the system how to automatically convert ILE profiles to get the geography data, and includes information on the mathematics and French. At the end of the integration phase, the teacher has a unique profile for each of his or her students.

The second phase of the environment proposes rich uses of thus created profiles. One of these uses is made by the Adapte module offering learners activities adapted to their profiles. These activities can be worksheets generated by the system or computerized activities managed by an external ILE. This is done through the Adapte module, which we will now describe.

1.3. Adapte, a module offering personalized activities

The role of the Adapte module is to provide learners with activities suited to their profiles. These activities can be paper and pencil exercises or computerized activities managed by an external ILE.

In the case of paper and pencil activities, Adapte generates a worksheet matching the profile of each learner. To do so, it creates tailor-made exercises to be included in the sheet and determines the size of the worksheet itself. It also provides the teacher with answers to the exercises contained in the sheet.

In the case of computerized activities, Adapte sets personalized sessions on an external ILE according to the learner's profile. To do so, it uses ILE exercise generators or chooses exercises in the ILE database. It also determines the order in which the exercises appear, their number and the duration of the session.

In this paper, we describe the design work carried out on the portion of Adapte proposing paper and pencil activities.

2. Generation of paper and pencil activities

In the context of Adapte, a paper and pencil activity is a worksheet to be printed. The exercises on the sheet can relate to several disciplines and this, whatever school grade it is used in. We will therefore explain how we have done an inventory of exercises proposed to learners by teachers of all subjects in elementary school, in college and in high school. Finally, we will show how we have implemented a computer system able to generate these exercises, using existing generators or not.

2.1. Typology of exercises

By studying the curriculum 2005-2006 published in the official texts of the French Ministry of National Education, Research and Technology, and then working with teachers in elementary schools, as partners in the PERLEA project, we have identified fifteen types of exercises that can be offered to a student, taking into account all subjects and levels. The identified typology of exercises is presented in Figure 1. Before formally defining the terms contained in this typology, we will use an example to understand it better.

Let us go back to our teacher's example. He wants to create exercises so that his students can work on the classification of relative numbers in math. To do so, he selects the pattern "Organization of elements" (see C in Figure 1), then refines his choice by selecting the operational pattern "Classify objects" (see C1 in Figure 1).

From this operational pattern, the system presents the teacher with an interface enabling him to specify the constraints of exercise generation. It is at this point that the teacher expresses his desire to work on relative numbers and specifies that the organization of these numbers must be in ascending order. He also specifies the constraints on the choice of numbers contained in the exercise (e.g. no fraction, no multiple of 10…). All these constraints are saved in an exercise frame (see Frame C1.001 in Figure 1). It is from this exercise frame that the system will generate the exercises contained in the personalized worksheets. Thus it will generate different exercises from the same frame of exercises.

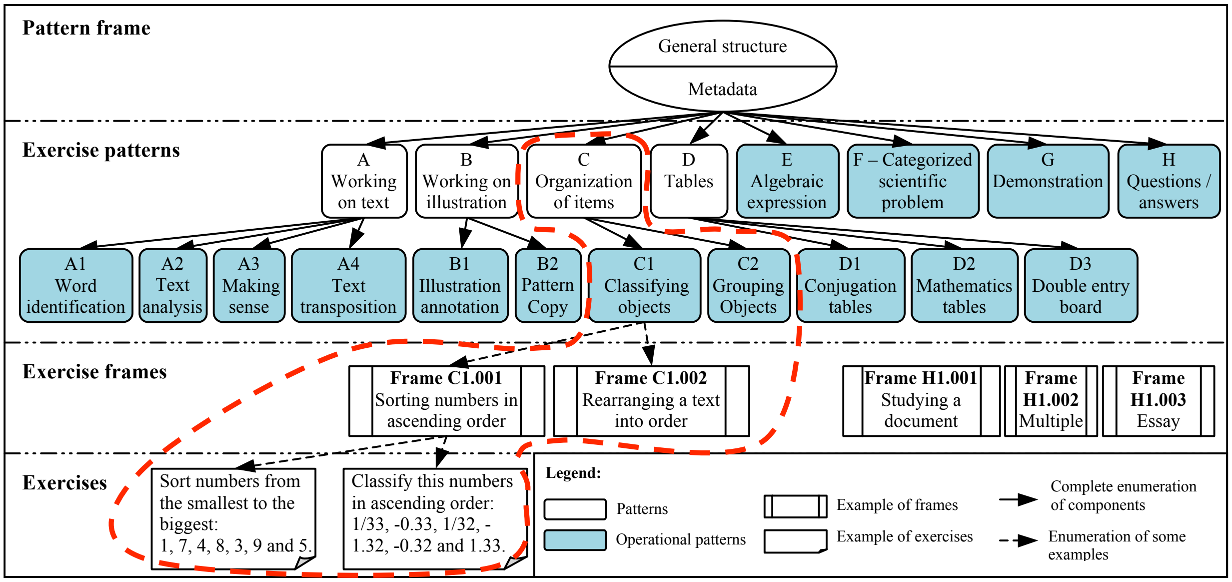

Figure 1. Typology of paper and pencil exercises

More formally, our typology contains eight exercise patterns, some of which can be broken up in several operational patterns. An **exercise pattern** (e.g. C - Organization of items, in Figure 1) defines a category of exercises generated with the same exercise generator. An **operational pattern** (e.g. C1 - Classifying objects, in Figure 1) specifies a subset of exercises generated through the pattern generator (here C), but with particular generation constraints. Our typology contains fifteen operational patterns to define fifteen types of exercises. The generic structure of these patterns and the set of metadata common to all patterns are defined in a pattern frame. From there, creating an **exercise frame** consists in associating an operational pattern with generation constraints. Creating an **exercise** consists in assigning to the parameters of the exercise frame values that respect these constraints. Thus created exercises will be composed of elements of wording and elements of answer to the learner, as well as the solution to the teacher.

2.2. Generation of exercises

In this section, we will present how the system generates exercises corresponding to the eight exercise patterns that we have identified in our exercise typology (see A to H in Figure 1). We will show how we studied existing generators, considered reusing some of them in our system and developed a generic architecture that we then specialized for generators useful to Adapte.

2.2.1. What type of generator for Adapte? The study of existing exercise generators leads us to sort them into three categories.

First, the **automatic generators** generate exercises without any intervention on the part of the user [1, 2]. Their advantage is to quickly create a large number of exercises, but they are not customizable by teachers. Therefore teachers can neither adapt them to their work habits, nor to the specificities of their students. On the opposite, the **manual generators**, named authoring tools, guide the user in the design of exercises [3]. Their advantage is to give the teacher complete freedom both in the application domain and in the educational content of the exercise. In return, the teacher must fully define the exercise and its solutions, which is long and tedious, and is an obstacle to the use of such systems. Half-way between these two types, the **semi-automatic generators** can construct the terms of exercises themselves, but they allow the user to intervene in the creative process by enabling him or her to specify a set of constraints on the exercise he or she wants to create [4]. The semi-automatic generators have the same advantages as automatic generators (quickly generating a large number of exercises) and provide a solution to their lack of flexibility: teachers can define the parameters of generated exercises.

In the case of Adapte, the most interesting seems to incorporate semi-automatic generators since they create a large number of exercises and enable the personalization of their generation. The problem is to know whether this is possible for all the exercise patterns that we have identified. For example, providing a semi-automatic generator for the creation of a Multiple Choice Test, for all domains and all levels, appears unrealistic. Indeed, it would use very large knowledge bases to cover all disciplines and all levels of education. A compromise could be to offer the teacher a manual generator enabling to provide the knowledge base for an exercise (a crop of questions with several propositions for each). Then, the system chooses several questions and answers, replacing some words by equivalent terms in order to diversify the wording. This solution enables, for the same exercise frame, to generate various Multiple Choice Tests: they will have no more than a few questions in common, and for these, they will not necessarily have the same proposed answers.

This approach, which consists in asking the teacher to provide the knowledge bases for the semi-automatic generators, is used in cases where classic semi-automatic generators seem unrealistic in our generic context.

Our choice being to use semi-automatic generators, we then discussed the possibility for each Adapte exercise pattern to use existing generators. If we except the exercises of type F in Figure 1, with the generators we have been able to study, either the teacher has to key in the exercises completely or he cannot intervene at all into the creation process. If we had used such generators, we would not have been able to offer teachers a random option in the generation of their exercises. For categorized scientific problems (see F in Figure 1), we have integrated GenAMBRE, the generator of AMBRE-Teacher [4], implemented to create aithmetic word problems in the ILE AMBRE-add. By providing the necessary knowledge bases, this generator can be used in a generic way and thus provide exercises on problems of combinatorial analysis, thermodynamics, etc.

2.2.2. Architecture of semi-automatic generators. To each exercise pattern presented in Figure 1 corresponds a generator that creates exercises for the learner and answers for the teacher. This answer will be either defined by the generator when it is possible, or keyed in by the teacher. Similarly, if some constraints are not specified by the teacher, they will be specified by the system. Moreover, at the moment of exercise generation, the exercise frame may contain constraints of re-generation preventing the same exercise to be generated again for the same exercise frame. All generators proposed for Adapte comply with a generic architecture (see Figure 2) that we will detail before illustrating it.

Thanks to this generic architecture of exercise generators, we can differentiate four levels. The **general level** contains the knowledge common to all domains for which we want to generate an exercise, e.g. the knowledge required to write a statement in natural language. The **domain level** contains the knowledge specific to the domain, e.g. the knowledge of calculation. The **generation level** contains the specific processes to create an exercise: definition of constraints on an exercise pattern saved in an exercise frame; instantiation of this frame to generate an exercise and its answer; layout enabling to provide exercises with a uniform presentation. Finally, the **exercise level** contains all the documents for the created exercise, including the exercise frame and its instantiation (wording of the exercise and its answer).

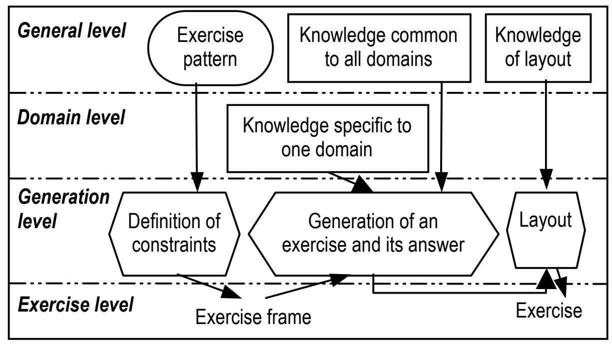

Figure 2 : Generic architecture of exercise generators

To illustrate the genericity of this architecture, we will explain how we have applied it to the categorized scientific problem generator (see Figure 3). This type of exercises, which can be found in scientific academic subjects (mathematics, physics, chemistry...) are based on classes of problems. We expect the student to solve the problem, for which we provided the question wording, by identifying the class of the problem. For example: "We have a deck of 32 cards. We get 5 simultaneously. How many draws contain exactly 2 knaves and 2 hearts?".To generate this type of exercise, the system has the exercise pattern corresponding to it, the knowledge of general level (grammatical knowledge, layout knowledge), as well as the domain knowledge (knowledge of the sentences, their theme and the possible complications of wording; knowledge of classification, reformulation and problem-solving techniques that enable the system to solve the problem). During the generation, the teacher specifies, thanks to an adapted interface, the generation constraints that are then saved in the exercise frame. The creation of the exercise uses the GenAMBRE generator [4] then the SYRCLAD solver [5] to provide a solution to the proposed problem. Finally, a layout phase standardizes the presentation of the exercises.

We specialized the generic architecture alike to define the exercise generators associated with the exercise patterns in Figure 1 except for the "Demonstration" pattern.

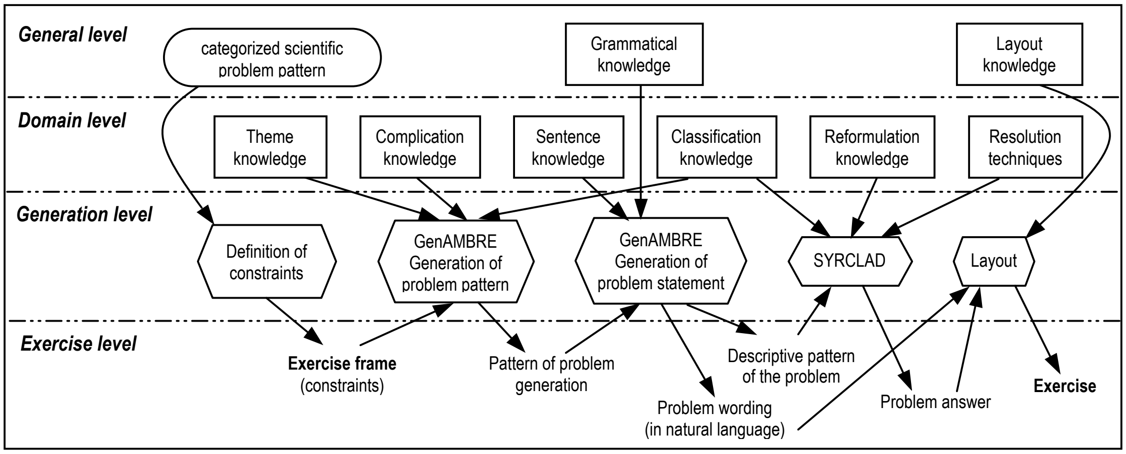

Figure 3. Architecture of generator for "Categorized scientific problems"

3. Validation

First, we proposed a typology of exercises that can be given to a learner at primary school and high school. This typology includes fifteen types of exercises. We defined it with the teachers associated to the PERLEA project. These teachers were primary school teachers. To test its scope, we have worked with secondary teachers. We watched each of the exercises they use for their French, English, mathematics, biology and history / geography classes and this for all levels of secondary school. All the exercises used were in our typology. Now, we must conduct a more rigorous evaluation with experts in educational sciences to validate our typology completely, both in its genericity and its completeness.

We then set the architectures of the eight exercise generators that we considered necessary to create exercises of our typology. These generic architectures can be used to develop exercise generators whatever context they are meant to be used in. If these architectures facilitate the generators setting up in new domains of application, the considerable work of instantiation of knowledge bases for a domain still remains unavoidable. We were able to test the genericity of these generators by implementing some of them in varied domains (e.g. we have implemented the table generator to propose conjugation exercises but also multiplication or addition ones, the “working on text” generator to make exercises in French, history etc.).

Then, we developed the Adapte software. The module design was made in partnership with teachers and, the software being now usable, we have submitted it to these same teachers. We also have submitted it to a teacher outside the PERLEA project. Their feedback seems to validate the software and its design. We must now make a more rigorous evaluation. This evaluation will be done by setting up experiments with teachers unrelated to the conception of the module. These experiments will involve all concerned modules of the PERLEA project environment, and range from the definition of a profile frame by the teacher to the effective use of personalized activities by learners.

In the continuity of this work, we are presently interested in the design of the second part of Adapte that will offer sessions suited to the skills of the learner on an external ILE.

4. References

[1] D. Bouhineau, A. Bronner, H. Chaachoua, and J.F. Nicaud, “Patrons d’exercices pour APLUSIX : Une étape du développement de l’EIAH occasion d’un travail entre didacticiens et informaticiens”, *EAIH'2005*, France, 2005, pp. 377-382.

[2] R.R. Burton, “Diagnosing bugs in a simple procedural skill”, *Intelligent Tutoring Systems*, Academic Press, London, 1982.

[3] J.P. David, A. Cogne, and A. Dutel, “Hypermedia exercises prototyping and modelising”, *Computer Aided Learning and Instruction in Science and Engineering*, 1996, pp. 252-260.

[4] N. Duclosson, S. Jean-Daubias, and S. Riot, “AMBRE-enseignant : un module partenaire de l’enseignant pour créer des problèmes”, *EAIH'2005*, France, 2005, pp. 353-358.

[5] N. Guin-Duclosson, “SYRCLAD : une architecture de résolveurs de problèmes permettant d'expliciter des connaissances de classification, reformulation et résolution”, *Revue d'Intelligence Artificielle*, 1999.

[6] S. Jean-Daubias and C. Eyssautier-Bavay, “An environment helping teachers to track students' competencies”, *Workshop LEMORE, AIED'2005*, Pays-Bas, 2005.