Generation of exercises within the PERLEA project

Stéphanie Jean-Daubias, Marie Lefevre, Nathalie Guin
Université de Lyon, France
Université Lyon 1 – LIRIS, CNRS, UMR5205
43 bd du 11 novembre 1918, 69622 Villeurbanne Cedex, France
{ Stephanie.Jean-Daubias, Marie.Lefevre, 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 are aiming at designing a module managing the generation of personalized activities. For this purpose, we suggested a typology of pencil and paper exercises that can be given to a learner, as well as the architecture of generators allowing the creation of all of these exercises. We also implemented and tested our proposition in a module helping the teacher to propose exercises suited to his students' knowledge.

Keywords. Interactive Learning Environments (ILE), personalization, generation of exercises, architecture, genericity.

1. Introduction

Personalization of learning is one of the major issues of Technology Enhanced Learning. Personalization relies in particular on using learners profiles to gather information about the learners, thus allowing to describe their knowledge, skills, perceptions and/or behaviors. These data are collected or deduced from one or several pedagogical activities, computerized or not 0.

Our approach consists in helping the teacher proposing to learners personalized pedagogical activities suited to their knowledge and gaps shown in their profiles, and suited to the teacher's needs and to the pedagogical context, expressed in what we name pedagogical strategies. To personalize pedagogical activities offered to the learner based on their profile, we can either use knowledge-based systems to generate the pedagogical activities best-suited to the profile, or provide the teachers with tools allowing them to perform this task themselves. We aim at linking these two options.

We first present the context of our research within the PERLEA project and its software environment. We then focus on the generation of exercises part in the Adapte module. To build this module, we first proposed a typology of exercises that can be given to a learner, together with the architecture of eight generators able to create all of these exercises. We detail these two aspects before moving on to their implementation and validation.

The PERLEA project aims at improving the integration of ILEs in education by building bridges between the use of ILEs and teachers' everyday practices. To do so, we are interested, in a generic way, in learners profiles and their a posteriori use for the management of learners and the personalization of learning 0. Hence we aim at developing an environment that would permit teachers to manipulate existing profiles. This environment consists of two phases: the integration of existing profiles (based on PMDL, the profiles modeling language that we proposed to unify external learners profiles to permit their reuse, either pencil and paper or software ones 0) and the management of thus unified profiles. The second phase of the environment proposes rich uses of the unified profiles. One of such uses is accomplished by the Adapte module, which offers to learners activities adapted to their profiles. These activities may be worksheets generated by the system or computerized activities to be done in an external ILE.

In the case of pencil and paper activities, Adapte generates worksheets matching the profile of each learner, according to teacher's pedagogical goals. To achieve this, it creates tailor-made exercises to be included in the sheets and determines the size of the worksheets themselves. It also provides the teacher with the answers to the exercises contained in the sheets. In the case of computerized activities, Adapte sets personalized sessions on external ILEs according to the learners profile. For this, it uses ILE exercises generators or chooses exercises in the ILE database. It also computes the number of exercises, in which order they appear and the duration of the session.

2. Generation of pencil and paper activities

For Adapte, a pencil and paper activity is a worksheet to be printed. The exercises on the sheet can relate to several disciplines, 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 from elementary to high school. Finally, we show how we have implemented a software able to generate these exercises, using existing generators when available.

2.1. Typology of exercises

By studying curricula published in the official texts of the French Ministry of Education, and subsequently working with teachers in elementary schools, as partners in the PERLEA project, we have identified fifteen types of exercises that can be proposed to a learner, taking into account all subjects and levels. The identified typology of exercises is presented in Figure 1.

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

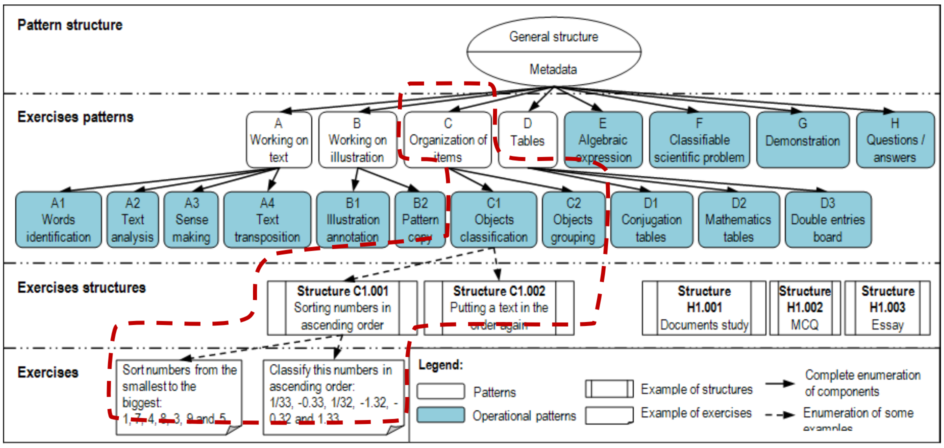

Figure 1: Typology of pencil and paper exercises

2.2. Generation of exercises

In this section, we present how the system generates exercises corresponding to the eight exercises patterns we have identified in our exercises typology (see A to H in Figure 1). We show how we studied existing generators, considered reusing some of them in our system and how we 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 exercises generators leads us to classify them into three categories.

First, fully *automatic generators* generate exercises without any intervention of the user [1] [2]. Their strength 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, *manual generators*, named authoring tools, guide the user in the design of exercises [3]. Their strength is to give the teacher complete freedom both in the application domain and in the educational content of the exercise. But, in return, the teacher must fully define the exercises and their solutions, which is a tedious task, that restraints the use of such systems.

Half-way between these two types, *semi-automatic* generators can construct the terms of exercises themselves, but they allow the user to intervene in the creative process by specifying a set of constraints on the exercises he wants to create 0. Semi-automatic generators have the same strengths as automatic generators (quickly generating a large number of exercises) and provide a solution to their lack of flexibility: teachers can tune the parameters of generated exercises.

In the case of Adapte, the most suitable approach is to incorporate semi-automatic generators since they create a large number of exercises and permit the personalization of their generation. The problem is to know whether this is possible or not for all the exercises patterns we have identified. For example, providing a fully generic semi-automatic generator for the creation of a Multiple Choice Questions test, for all disciplines and levels, appears unrealistic. Indeed, this would require using 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 him to provide the knowledge base for an exercise (a set of questions with several choices for each). Then, the system would choose several questions and answers, replacing some words by equivalent terms in order to diversify the wording. This solution allows, for the same exercises structure, to generate various Multiple Choice Questions tests: they will have no more than a few questions in common, and for them, they will not necessarily have the same proposed answers. We use this approach, which relies on the teacher to provide the knowledge bases for the semi-automatic generators, in cases where state of the art semi-automatic generators seem unrealistic in our generic context.

Having choosing to use semi-automatic generators, we then studied the possibility for each Adapte exercises pattern to use existing generators. If we except the F-type exercises of Figure 1, with the generators which were available to us, the teacher has either to key in the exercises completely or he cannot influence at all the creation process. Using such types of generators would have prevented us to propose a random option to teachers in the generation of their exercises. For categorized scientific problems (F-type in Figure 1), we integrated into Adapte, GenAMBRE, the generator of AMBRE-Teacher 0, implemented to create arithmetic word problems in the AMBRE-add ILE. By providing the necessary knowledge bases, this generator could 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 exercises pattern presented in Figure 1 corresponds a generator that creates exercises for the learner and answers for the teacher. An answer will be either defined by the generator if possible, or keyed in by the teacher. If some constraints are not specified by the teacher, they will be specified by the system. Moreover, at the time when an exercise is generated, the exercises structure may contain constraints of re-generation preventing the same exercise to be generated again for the same exercises structure. All generators proposed for Adapte comply with a generic architecture (see Figure 2) that we will detail before giving an example. The knowledge of the generators is provided partly by the designer of the system, and partly by the teachers who thus complete gradually the knowledge bases.

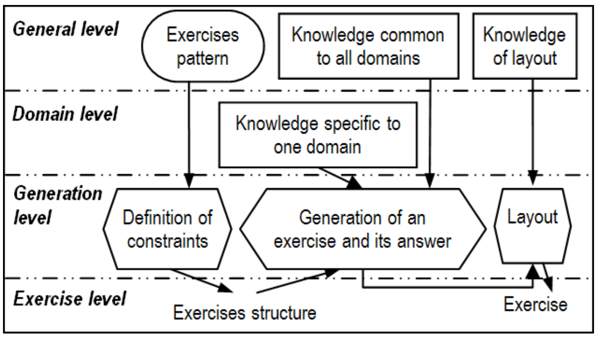

Figure 2: Generic architecture of exercises generators

Thanks to this generic architecture of exercises generators, we can specify 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 exercises pattern saved in an exercises structure; instantiation of this structure 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 exercises structure and its instantiation (wording of the exercise and its answer).

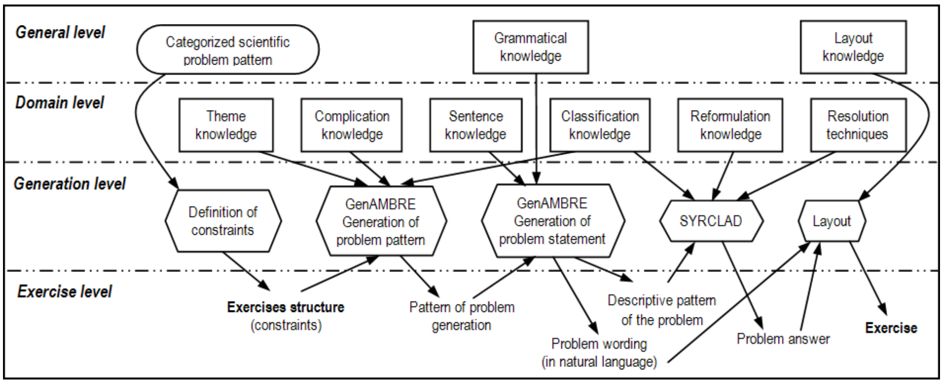

Figure 3: Architecture of generator for "Categorized scientific problems"

To illustrate the genericity of this architecture, we 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...), is based on classes of problems. We expect the student to solve the proposed problem by identifying its class. For example: "We have a pack of 32 cards. We take 5 simultaneously. How many draws contain exactly 2 knaves and 2 hearts?". To generate this type of exercises, the system has the exercises pattern corresponding to it, the knowledge of general level (grammatical knowledge, layout knowledge), as well as the domain knowledge (knowledge on 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). Before the generation, thanks to an adapted interface, the teacher specifies, the generation constraints that will be saved in the exercises structure (for example the problem class, the characters or objects of the wording). The creation of the exercise uses the GenAMBRE generator 0 then the SYRCLAD solver 0 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 exercises generators associated with the exercises patterns except for the "Demonstration" pattern.

3. Implementation

These architectures are implemented in the Adapte module. When a teacher wants to create an exercise, he has first to choose the corresponding type of exercises on the screen presenting the different exercises patterns (cf. Figure 1 and Figure 4). For example, to propose conjugation exercises to his learners, the teacher chooses the “tables” pattern and the operational pattern corresponding to conjugation.

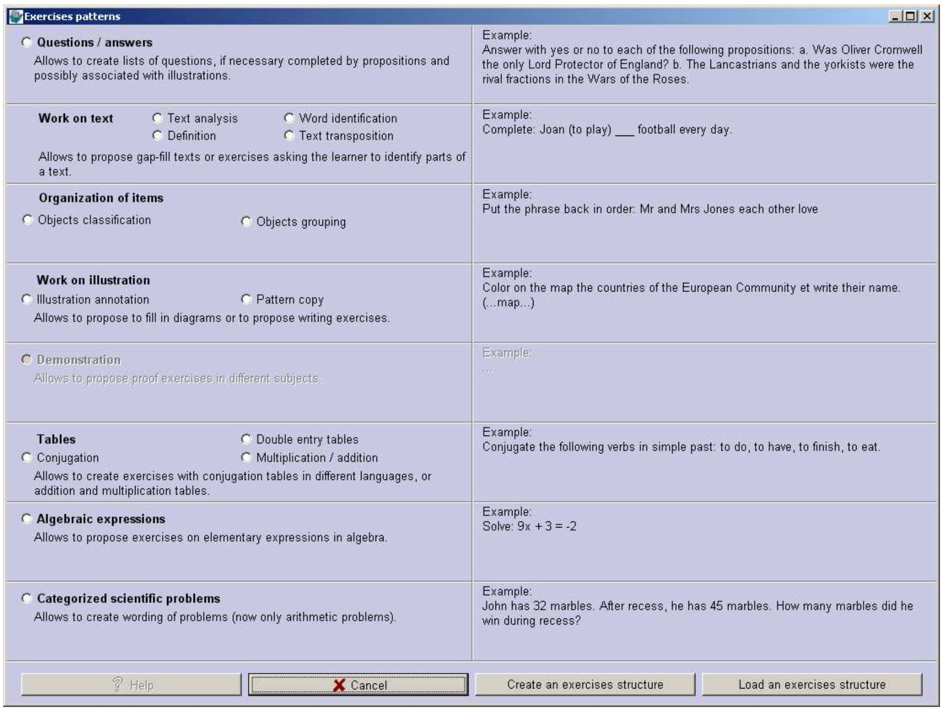

Figure 4: Choice of exercises patterns in Adapte.

From this operational pattern, the system presents the teacher with an interface enabling him to define the constraints of exercise generation depending on his pedagogical goal (cf. Figure 5). The teacher chooses a language (English in our example) and can specify the tense (indicative present in the example), persons (random in this example), types of verbs (regular or irregular for English language) and/or verbs (to eat and to play are priority verbs in the example), and the number of verbs to be proposed to the learners (for example 2 priority verbs among 5 verbs).

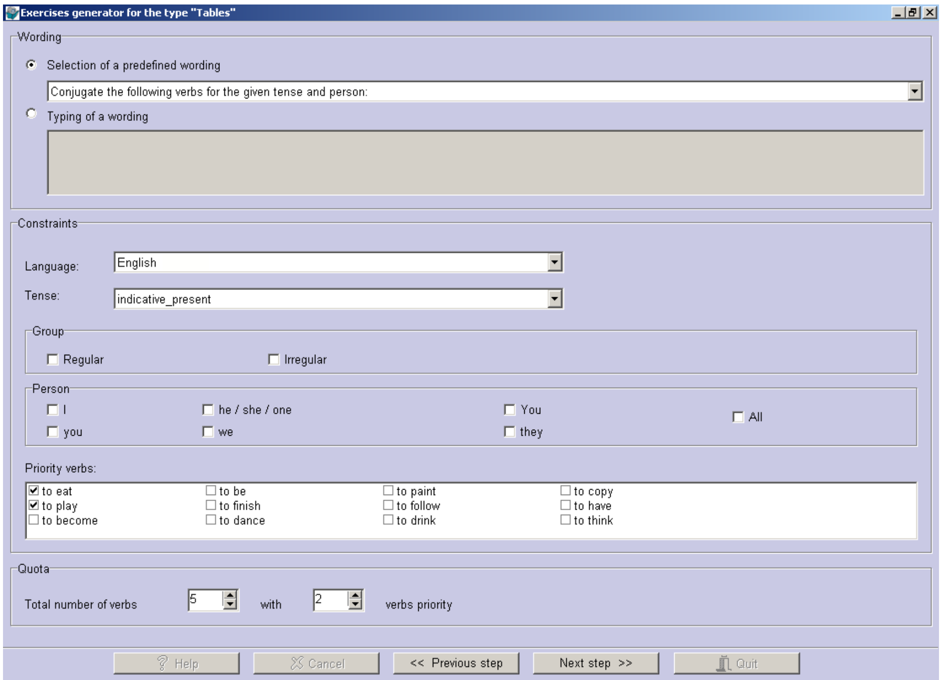

Figure 5: Definition of constraints for the "conjugation" operational pattern of Adapte.

Wording: Conjugate the following verbs for the given tense and person: to eat,to play,to become,to follow,to finish.	to eat : I eat, you eat, he eats, we eat to play : I play, you play, he plays, we play to become : I become, you become, he becomes, we become to follow : I follow, you follow, he follows, we follow to finish : I finish, you finish, he finishes, we finish
Wording: Conjugate the following verbs for the given tense and person: to eat,to play,to paint,to have,to follow.	to eat : we eat to play : we play to paint : we paint to have : we have to follow : we follow
Wording: Conjugate the following verbs for the given tense and person: to eat,to play,to be,to finish,to dance.	to eat : I eat, you eat, we eat, you eat to play : I play, you play, we play, you play to be : I am, you are, we are, you are to finish : I finish, you finish, we finish, you finish to dance : I dance, you dance, we dance, you dance

Figure 6: Example of exercises generated by Adapte from the previous constraints.

All these constraints are saved in an exercises structure, described with metadata to facilitate its reuse. The system generates the exercises contained in the personalized worksheets from this exercises structure. Thus it generates different exercises from the same exercises structure (cf. Figure 6).

4. Conclusion

As a solution to the problem of personalization of learning, we established an approach helping teachers to propose pedagogical activities suited to learners' knowledge and to teachers' needs. In this framework, we focused here on the generation of pencil and paper exercises.

First, we presented our typology of exercises that can be given to a learner from primary school to high school. This typology includes fifteen types of exercises. We defined it with the primary school teachers associated to the PERLEA project. To test its scope, we have worked with secondary teachers. We observed each of the exercises they use for their French, English, mathematics, biology, history and geography classes for all levels of secondary school. All the exercises used were in our typology. Now, we have to work with experts in educational science to completely validate our typology, both in its genericity and its completeness.

We then proposed a generic architecture of exercises generators and set the architectures of the eight exercises generators that we considered necessary to create the exercises of our typology. These generic architectures can be used to develop exercises generators whatever context they are meant to be used in. If these architectures facilitate the setting up of generators in new domains of application, there is left to do a considerable work of instantiation of knowledge bases for a new domain. We were able to test the genericity of these generators by implementing some of them in varied domains (e.g. we have implemented the tables 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 Adapte. The module design was made in partnership with teachers (according to differentiated design 0) and we have submitted it to these same teachers and to a teacher outside the PERLEA project. Until now all their feedbacks seem to validate the software and its design. The system is usable and permits teachers to define the constraints allowing to generate exercises matching their needs (expressed in the constraints they defined) and their learners' knowledge (due to Adapte functionalities not presented in this paper 0). We must now make a more rigorous evaluation. This evaluation will be conducted with experiments with more teachers unrelated to the conception of the module. These experiments will involve all concerned modules of the PERLEA project environment 0, and range from the description of a learners profiles structure by the teacher 0 to the effective use of personalized activities by learners 0.

In the continuity of this work, we are presently finishing the implementation of the Adapte part that offers sessions suited to learners' skills on an external ILE.

5. References

Bouhineau, D., Bronner, A., Chaachoua, H. and Nicaud, J.-F. (2008). Helping Teachers Generate Exercises with Random Coefficients. International Journal of Continuing Engineering Education and Life-Long Learning, vol 18-5/6, 520-533.

Burton, R. R. (1982). Diagnosing bugs in a simple procedural skill. Intelligent Tutoring Systems. London, Academic Press, 157-184.

David, J.-P., Cogne, A. and Dutel, A. (1996). Hypermedia exercises prototyping and modelising. Computer Aided Learning and Instruction in Science and Engineering. S. B. Heidelberg: 252-260.

Guin-Duclosson, N. (1999). SYRCLAD : une architecture de résolveurs de problèmes permettant d'expliciter des connaissances de classification, reformulation et résolution. Revue d'Intelligence Artificielle, vol. 13-2, éditions Hermès, 225-282.

Jean-Daubias, S. (2009). Differentiated design: a design method for ILE. Research report RR-LIRIS-2009-015.

Jean-Daubias, S. and Eyssautier-Bavay, C. (2005). An environment helping teachers to track students' competencies, Workshop Learner Modelling for Reflection, Artificial Intelligence in Education (AIED'2005), Netherlands, 19-23.

Jean-Daubias, S., Eyssautier-Bavay, C. and Lefevre, M. (2009). Harmonization of heterogeneous learners profiles, Research report RR-LIRIS-2009-013.

Jean-Daubias, S. and Guin, N. (2009). AMBRE-teacher: a module helping teachers to generate problems, 2nd Workshop on Question Generation, Artificial Intelligence in Education (AIED'2009), Brighton, Great Britain.

Lefevre, M., Cordier, A., Jean-Daubias, S. and Guin., N. (2009). A Teacher-dedicated Tool Supporting Personalization of Activities, World Conference on Educational Multimedia, Hypermedia & Telecommunications (ED-MEDIA 2009), Honolulu, Hawaii.