AIED 05 WORKSHOP 11

Amsterdam, 18-22 July, 2005

Learner Modelling for Reflection, to Support Learner Control, Metacognition and Improved Communication

AIED 05 WORKSHOP

12th International Conference on Artificial Intelligence in Education, Amsterdam, the Netherlands

Preface

Learner modelling is at the core of AIED research, as the learner model is the foundation of *system that care* because they have the potential to treat learners as individuals. This is reflected in the proportion of AIED and ITS research that has a focus on learner modelling.

This workshop brings together researchers working towards the emerging roles for learner models. This means that learner models go beyond their role in personalising teaching,an addition to their essential role for personalising teaching, becoming first class objects which are available to students and teachers as a basis for improving learning outcomes. Importantly, open learner models offer the potential to help learners and teachers to reflect on their own knowledge, misconceptions and learning processes.

At one level this workshop follows on from previous workshops, the AIED 1999 *Open, Interactive and Other Overt Approaches to Learner Modelling*, held at AIED 1999 in Le Mans, France, the ITS 2002 workshop I*ndividual and Group Modelling Methods that Help Learners Understand Themselves*, held in San Sebastian, Spain, and the AIED 2004 workshop *Learner Modelling for Reflection* 20 July 2003, Sydney, Australia. It is also core to the Learner Modelling for Reflection (LeMoRe) group, who have open learner models as one of their research interests. This workshop also moves us to important new ground, reflecting some of the important new possibilities and demands of open learner modelling and new roles for learner models.

A particularly important new direction is to incorporate open learner models into conventional learning systems since these are widely deployed and maintain large collections of data about individuals. The challenge is to fruitfully make this data more useful as detailed models of learner development, with modelling of competence, knowledge and other aspects. To meet this challenge, we need to explore ways to support the classroom teacher in defining learner models, easily making the connection between available data and the learner model and then providing interfaces that will enable learners and teachers to really see the learner models in ways that will support reflection.

A closely related area of importance is how best to collect, analyse and externalise data from learner interactions and how to represent this for most effective support of reflection. What is the role of an episodic learner model? A summative model? A comparative model? Can we link the model available to the direct actions and activities that the user may recall? How can we do this effectively to enhance metacognitive processes such as awareness of learning style?

A hot topic for the role of learner modelling for reflection involves taking proper account of the social context and capturing this effectively in the learner model. This includes the support of collaborative learning where we need to develop the potential of open models to improve communication, collaboration and cooperation between students who are learning together. It also includes the role of comparative models, where the learner sees not just their own, absolute learner model, but may also be allowed to see how this compares with that of other students. For example, a student who aspires to perform at the highest standard may be interested, even motivated by knowing how their learner model compares with that of students who are high achievers, or, based on historical data, the student may be able to see their own performance against that of groups from previous years. There is also the rather interesting case of the student who wants to learn precisely what is essential to pass and not more.

This leads immediately to another critical direction, the need to establish methods for measuring the benefits of reflective learner models. For too long, we have relied on existing educational research which indicates that reflection is important for improved learning outcomes and yet, the cost of measuring differences in outcomes is often prohibitively high. However, we need ways to determine effectiveness of open learner models for reflection, both to ensure our instantiations of open models are actually delivering the promised benefits and so that we can start to improve our understanding of just what types of learner models are really of value for improving learning. For example, is a single, simple skillometer just as effective as a large detailed student model? Does the opportunity to interact with the model make for improved learning outcomes? If so, which aspects of the interaction seem most important?

There are many challenges in building and evaluating effective interfaces to learner models. New hardware creates new opportunities. There is the potential to explore ways to make large and complex models accessible and open. With the growing possibilities of just-in-time learning and ubiquitous learning, there are new challenges for building user interfaces for open learner models.

Open learner models can serve as a focus for mixed-initiative interaction where the learner can be challenged about their learning and where the learner can also challenge the system about their learner model. There is a considerable body of work on self-explanation and its benefits and we need to explore how this can best be linked to effective opening of learning models. Equally, the right interface might help students discuss their learner models with peers and teachers. We need to gain greater understanding of the ways that such interactions can be supported.

The title of this workshop also refers to improved communication between teachers and learners. This is a particularly important direction for open learner modelling because it is widely acknowledged that a common reason for failures in teaching are due to miscommunication from the teacher about the intended learning goals and what it means to achieve them as well as how to go about achieving them. One approach to improving this situation is to use criterion-based assessment and to take care to provide the learners with detailed meaningful descriptions of the criteria. We need to explore how this can be linked with open learner modelling where the system often encodes a detailed operational form of this information within the student model and the modelling processes. Equally, learners may benefit from being able to communicate their understanding of the learning goals to the teacher with the aid of an open learner model. Not every student has the same aspirations and motivations for undertaking a learning activity. Can open learner models help the learner see that their aspirations are being taken into account adequately? Can teachers also learn from this, for example, tuning the personalisation to better meet the needs of the student who has the goal of just gaining a qualification? Or the student who does not care about grades, but wants a deep understanding? Or the many other students with the different goals?

Another important new direction for open learner models is in the support of learner control over learning. For example, how can we externalise learner models effectively so that learners can determine how they are doing and can initiate planning of their future learning activities. For example, can we enable a learner to determine that they have done enough practice on one concept and are performing well enough to meet the standard they expect of themself and can safely move on to new learning areas? Can we cater for different standards of learning that different learners want to achieve, so that, for example, the student who simply wants to earn a bare pass is supporting in achieving that goal?

At quite a different level, we are seeing the emergence of systems that model affective aspects such as emotion. We need to link this with the potential role of open learner models. How can we help students to understand how their emotions and attitudes may affect their learning? Can this also help teachers interpret student achievement and interaction patterns? Since the modelling of affective aspects seems likely to be especially error prone and may be construed as particularly sensitive aspect, can open learner models play an especially important role in improving the accuracy of the learner model and ensuring that the learner is aware of how it is used.

Finally, there is considerable work in machine learning in conjunction with learner modelling. This is often predicated on the assumption that a machine learning system can access collections of student models. Should we support learners in deciding just which parts of their learner models should be available to such processes? If so, this seems to require open learner models that learners can interact with to mark those aspects that are to be available.

The workshop had 15 submissions. Each of these was rigorously reviewed by three members of our expert international panel of reviewers on the Programme Committee. We have selected the eleven as full papers and one short paper in these proceedings.

We thank the members of the Programme Committee for their important contributions to the workshop, providing major input to the formulation of the workshop goals and scope and helpful feedback on submissions and careful reviews.

Judy Kay
Andrew Lum
Diego Zapata-Rivera

Programme Chairs

Judy Kay, University of Sydney, Australia
Andrew Lum, University of Sydney, Australia
Diego Zapata, Educational Testing Service, USA

Programme Committee

Susan Bull, University of Birmingham, UK
Paul Brna, Northumbria University, UK
Peter Brusilovsky, University of Pittsburgh, USA
Al Corbett, Carnegie Mellon University, USA
Vania Dimitrova, University of Leeds, UK
Jim Greer, University of Saskatchewan, Canada
Gord McCalla, University of Saskatchewan, Canada
Rafael Morales, Northumbria University, UK
Kyparisia Papanikolaou, University of Athens, Greece
Nicolas Van Labeke, University of Nottingham, UK

Table of Contents

Reactions to Inspectable Learner Models: Seven Year Olds to University Students

Susan BULL, Manveer MANGAT, Andrew MABBOTT,
Abdallatif S. ABU ISSA & Josie MARSH

Electronic, Electrical and Computer Engineering,
University of Birmingham, Edgbaston, Birmingham B15 2TT, U.K.
s.bull@bham.ac.uk

Abstract. This paper examines student reactions to inspectable learner models. We look at a simple example for children as young as 7, and university students using more complex inspectable learner models - one with multiple views on the model data, and one that can be opened to peers and instructors. We provide a descriptive account of student perceptions of their learner models, to complement the more formal data available elsewhere. This information is useful to those considering opening learner models in their systems, as it provides greater insight into individual student attitudes, which is important when supporting individual learning.

1. Introduction

One of the main educational aims of opening the learner model to the student modelled, is to facilitate reflection on learning, thereby supporting the learning process. There are now increasing numbers of systems employing open learner models (OLM) of various kinds. The presentation of models ranges from skill meters [1,2], to more detailed presentations such as hierarchical tree structures [3]; conceptual graphs [4]; textual explanations [5]. However, as yet we know very little about what students think of open learner models. This paper presents a descriptive account to researchers and developers considering opening the learner models of their systems, about what learners really think about the utility of OLMs, and their likelihood of using them. For example, if it is essential that a learner view their learner model in order for a system to support learning in the manner intended, it is necessary to know at an individual level, whether students would take part, and why and how. Such descriptive information is often obscured in more formal evaluations.

OLMs can be inspectable, editable by the user, or negotiated by student and system (see [6]). In this paper we concentrate on 3 inspectable learner models: Flexi-OLM which offers university students a choice of 7 views of their model; UMPTEEN which allows university students to not only view their own model, but also to open it to peers and instructors, and to view peer models that have been opened to them; and Wandies which has a simple OLM for 7-8 year-olds. Student reactions to these OLMs are described.

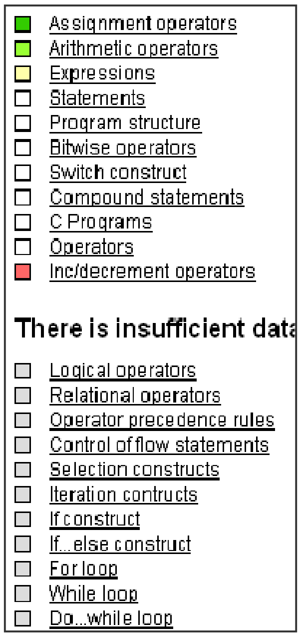
2. What Students Want in an Inspectable Learner Model

A previous survey found that university students might be receptive to using an OLM [7], but that survey was carried out amongst students who had not yet used one. The same survey has since been undertaken with a group of 67 MSc and 3rd year undergraduate students who had used Flexi-OLM. Results indicated that students wished to be able to access representations of the concepts they understand (54 students); problematic areas (62) and misconceptions (58). The latter is particularly interesting as this is something that is often not explicitly provided in more standard forms of feedback. Students further stated that they want to access their model to help them think about their difficulties to solve their problems (47) and, to a lesser extent, to plan their learning (39) and gain control over the learning process (39). Thus students can be receptive to using an OLM both before trying it (important for their motivation to use a system), and also after having experienced one.

3. The Inspectable Learner Models Used by Students

The students who used Flexi-OLM and UMPTEEN were studying for degrees such as Human-Centred Systems; Computer Interactive Systems; Communications Engineering; Electronic and Computer Systems Engineering - in an Electronic, Electrical and Computer Engineering department. The children who used Wandies were randomly selected from two classes in a junior school having children with a range of abilities and backgrounds.

Flexi-OLM has an OLM (for C programming) that can be viewed in 7 formats, according to the user's preference: alphabetical index, list ranked according to knowledge, concept map, hierarchical structure grouping related concepts, pre-requisites structure, lecture structure, textual summary. In all but the summary, coloured nodes indicate the level of understanding, using shades from pale yellow to dark green; and red to show the existence of misconceptions. Misconceptions are described textually in each view. Three of the views are shown in Figure 1 as an example, using the ranked list, lecture structure and pre-requisites structure.



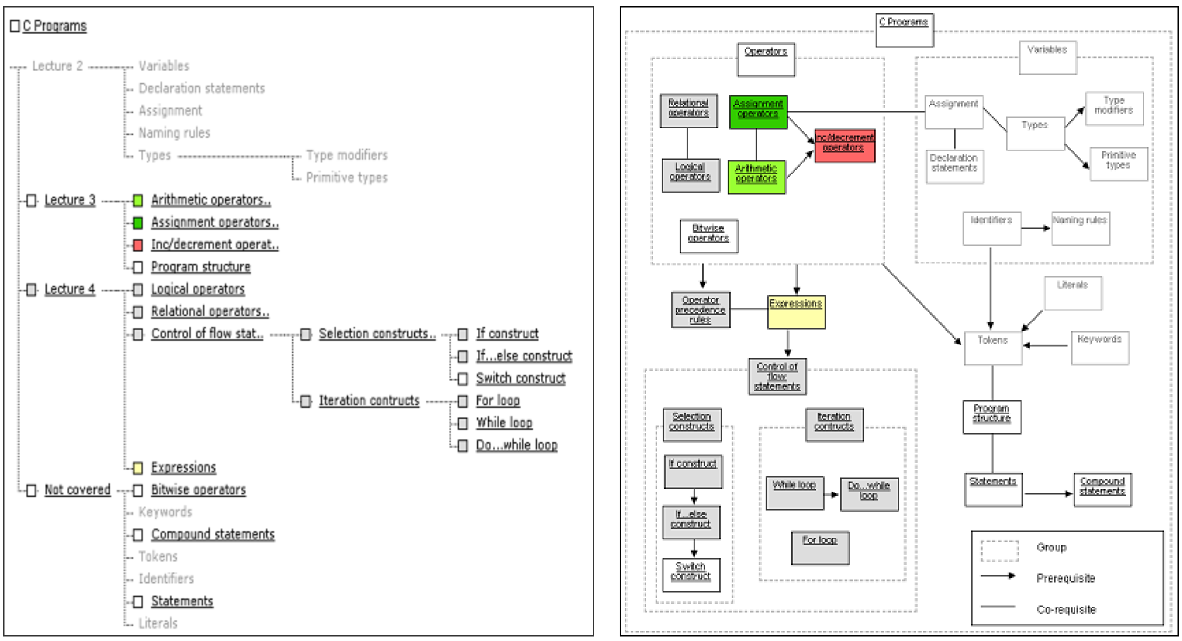


Fig. 1. Multiple views of the learner model (university students)

A study of an earlier version of the system with 4 views of the learner model, found that students had different preferences for which view to use, but none of the views was considered the most or the least useful overall [8]. It was therefore recommended that a range of views should be provided. The current version was evaluated with 39 MSc students in a laboratory setting. Students still found the availability of multiple views useful (7 choices for accessing their model was not overwhelming). Post-interaction questionnaires showed that each view had between 24 and 31 students finding it useful, with only between 2 and 4 giving negative ratings for any view. 1 user liked no views; 9 would use mainly one (but different views were preferred); 10 would mostly use two views; 7 would use three views regularly; 5 would use four views; 3 would use five views; and 4 would use all seven. Thus 29 of the 39 students expected to use multiple views, 19 of these using three or more views. The combination of views preferred differed, further suggesting maintaining the availability of all options.

UMPTEEN (User Models for Peers and Teachers to Encourage Emulation and Networking - for C programming) has an OLM that allows a user to see a statement of topics known and to what extent, associated skill meters, and descriptions of misconceptions (Figure 2). Students may open their model to other students and instructors named or anonymously, to all or selected participants, and can view models that have been opened to them. Thus users have access to multiple (or umpteen) models. Individual peer models are presented in the same way as one's own model. Group statistics are also available, listed by topic, with number of students with good, fair and weak knowledge, and misconceptions, of each topic (see [9]).

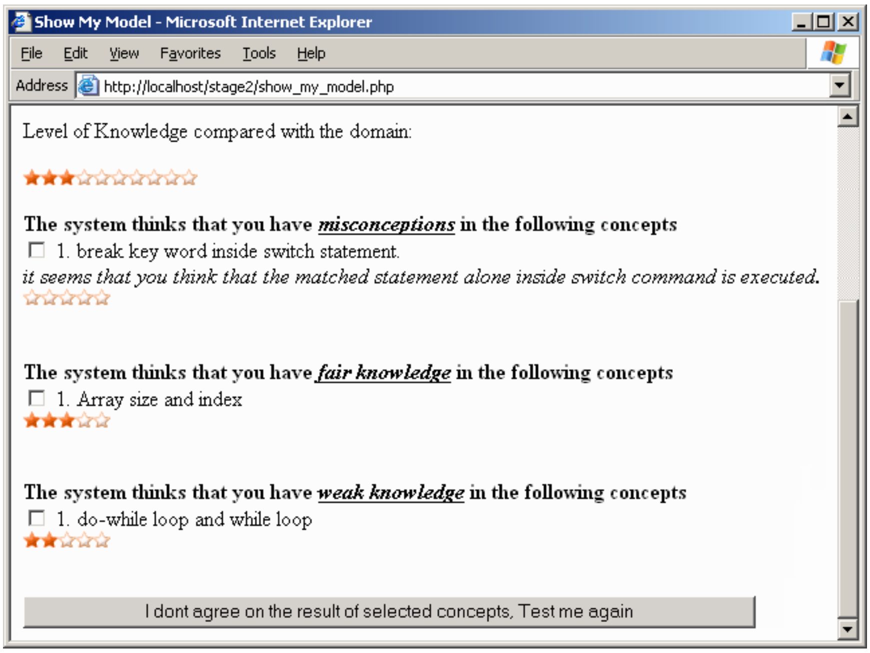


Fig. 2. A learner model that can be opened to peers and instructors (university students)

Previous work found that amongst a small group (12 users), most opened their learner model to all peers, half anonymously and half named [9]. In the current study, 50 MSc students used UMPTEEN over two lab sessions. 10 kept their learner model hidden from other students. 10 opened their model to everyone (2 named, 8 anonymously). 6 opened their model to at least ten peers; 9 opened their model to between five and nine people; and 15 opened their model to between one and four others. Some opened their models with their names, and some anonymously. Those who opened their model to only a few, tended mainly to open it to closer friends, with their names. 28 students opened their models to instructors (15 named and 13 anonymously). Students seemed to be using UMPTEEN in different ways - all of which are supported. Collaboration amongst friends can be facilitated by sharing models with a few people; finding suitable learning partners or broader networking can be facilitated amongst those who open their models more widely (as shown below, students often opened their model to those who opened their models to them); and individual reflection and competition can be supported for all, regardless of whether they choose to open their own model to peers.

In contrast to the above, the third OLM is very simple, for use by children. Previous results for an OLM with 8-9 year olds, using smiley faces to represent knowledge, were quite positive [10]. Here we consider even younger users, and an even simpler learner model presentation. Wandies Magical World of English supports classroom work on literacy for 7-8 year olds, following the requirements of the U.K. National Curriculum. The interaction uses the theme of wizardry to sustain interest and motivation. This theme is maintained in the OLM, where knowledge level is presented in the form of magic wands - gold for the top level (whizzing wizard), then silver (good wizard), and bronze (trainee wizard). These categories follow the Olympic medal colours, and are familiar to the children as they are also used in the classroom. Red represents the lowest level of understanding (students do not yet understand the spell). White indicates that a topic has not been attempted. Moving the mouse over a wand provides a little more information. This approach of using colour aims to facilitate transition to more complex OLMs that also use colour to represent level of knowledge, as the children grow older and can use more complicated learner model presentations. Wandies' simple OLM is illustrated in Figure 3 (the example of more detailed information given for the Magical Marks gold wand is: 'you understand this subject and can use the rules fully').

Fig. 3. A very simple OLM (7-8 year olds)

An initial evaluation was performed with 15 7-8 year olds, 5 working individually and 10 in mixed-ability pairs. Children used Wandies twice. In the paired condition, the learner model reflected the knowledge of the pair. Most children were engaged in the learning process, in both the individual and paired condition. There was an average 9% increase between pre- and post-test for those working individually, and 18% for pairs (tested individually). Thus it was not the case that the OLM was useful only to the stronger child, but all paired students appeared to be learning from discussion, and agreeing on answers. Children were observed to help each other to a greater extent than usual when they saw someone had red wands - i.e. assisting others occurred not only within pairs for those with a partner, but also more widely.

The purpose of this paper is not to present formal evaluations of these systems, but to investigate student reactions to various kinds of OLM. The aim is to provide descriptive data for researchers and developers, of student perceptions of inspectable learner models, and how they might be used. This kind of data is often not available in more formal evaluations.

4. (Why) Do Students Want Inspectable Learner Models?

Kay speculates that learners might have a variety of reasons for using OLMs, including: understanding the extent of their knowledge; increasing their awareness of what they know; comparison of their understanding to that of peers; appreciating how much experts know [3].

These and similar questions are the kinds of issue that we investigate in the context of users who have had experience with OLMs of different types. Students submitted written descriptions of the utility to their learning, of OLMs, based on their experience of the systems described above. The aim was to discover whether students understood the role of an OLM, by allowing them to explain freely, what they got (or did not get) out of using them. Beyond requesting a description of how the OLMs affected their learning, students were not further guided on what to write. The excerpts below cover all issues arising. (Note: some students are not native English speakers. Language errors have been preserved in the excerpts.) In the case of Wandies where, due to their age, children were not asked to make written descriptions, the information was taken from observer notes, and discussions with the two class teachers.

Viewing the learner model (UMPTEEN)

- The learner model showed a direct picture of my C ability, this is really helpful. I can arrange my future study in C programming.
- The list of my understanding for each topic of C programming helps me to remind my weaknesses and strengths next day.
- I opened the learner model straight away because I was interested to see how I'd done in the questions. When you normally begin to learn about a subject you don't really know how much of the content you understand. I like to know how I'm doing in my work (what level I'm at). For me being able to view the OLM is a confidence booster.
- After viewing my open learner model, I know I have weak knowledge in array size and index, so I think viewing my learner model is useful, but not so strongly, because when I choose which parts to be tested, I refuse to choose two parts, only chose the other parts I thought I am good at, aim to get reasonable mark. So I can not know my learning progress on a whole clearly.

Generally, students claimed positive effects for viewing their learner model, as illustrated in the first three excerpts above. They perceive benefit in planning their learning as the OLM helps them to identify where further work is required, and can also act as a reminder when they return to a system for subsequent interaction. The third excerpt illustrates in addition, the eagerness and motivation experienced by many, to access this information. It can also help confidence. However, benefits can only arise if the system is used in a serious attempt to support learning. The fourth excerpt shows that some students may actually learn little precisely because there is an OLM available. In this case this may have been because the user did not wish to receive negative feedback, or it may have been that he wished to open a good learner model to others, to appear better. How to deal with such cases is a difficult question. However, this was the only example in the data, contrasting with many positive comments.

Viewing multiple representations of the learner model (Flexi-OLM)

- Related concepts and pre-requisites are clear to show the outline knowledge. It is very useful to study logically. Lectures is also very important for student to see which part of course is poor for them. And to help them study all around.
- I really liked the ability to view different representations of knowledge.
- The concept map was the most useful as it shows the relationship between all subject areas and where my weaknesses lie.
- Concept map is a bit complex compared to the others, making it a bit difficult to understand.
- Pre-requisites is useful. I would spend more time using this view.
- I don't understand very well pre-requisites.

In general students found it useful to have multiple views of their learner model. As stated previously, there were no views preferred by all or most students, and no views disliked by the majority. Some students favoured a single view, whereas others liked having multiple representations. Individual differences in preferences are illustrated in the above comments where it is shown that some students liked the complexity of the concept map or pre-requisites view, while others found this complexity more confusing, preferring alternative views. This suggests the importance of providing multiple representations of learner model content, in order that each individual can use the view(s) that are most useful for them.

Viewing peer models (UMPTEEN)

- Comparing with others can let us know which level we are in. Is my performance much better than I supposed, or although I did well, I just reached the average level?
- I feel as a person that I'm quite competitive. So the opportunity to see how my peers had faired was something I really wanted to do. It was interesting to compare my worst subject areas with other people's to see if they had struggled with that area. My first feeling was not to try and get the highest mark but to get above average (this is the first figure that really meant anything to me), these figures helped to provide goals.
- Viewing the learner model of my peers enabled me to compare my progress. My aim after the comparison was to improve myself, and to be up there with my colleagues, and the open learner model provided an interesting and motivating environment to achieve that.
- Viewing the group model let me know not only one out of three students have problems on concept of array size and index, it let me to realize that I am performing better than I thought, low mark on this concept is not all my fault, it is actually quite difficult for students, what I need to do is do not lose my courage and confidence, study hard.
- Viewing their learner models with details let me know who has the same misconception as me and who is better. I consulted them and discussed together. For example, xxx has more experience in array size and index and he told me some details in array concept and boundary situation, which is my misconception. After that, we answered questions again and each of us felt this intercommunication was more effective than usual.
- When I click on the person's name xxx, I find he has similar knowledge in the concepts of pointers and addresses. Then we can help each to discuss this concept, which improve both of our knowledge. And when I open the person named yyy, he has better knowledge than me in the concept of bitwise and logical operators. I can learn more from him. When someone knows my strong parts, they can get some help from me.
- If my knowledge is under the average level of my class, I will get a lot of pressures, then, I will study harder and practice more than before. This is good for my learning.
- I did not understand how learner model of others could help my learning at the start, however, I have found learner model of others useful to recognise my strengths and weaknesses.
- Looking at average student model insisted me to stop looking for more knowledge since, being above the average give me a satisfaction feeling.
- In the situation that the most students are better than me in their learner model, I think viewing others' models will not help me a lot because this may do harm to my self-confidence and lead me to give up learning if there is a big gap between others and me.

Most students found it useful to view the learner model of others. A common comment was that this helped them to appreciate their position in the class, and the comparison of their strengths and weaknesses to others - which was more helpful than standard feedback. It also enabled them to better judge whether topics were generally easy or difficult and, in the latter case, weak performances resulted in fewer instances of low confidence. Most students found the peer models motivating, and used them to try to improve their performance where they perceived comparative weakness. Many reported using peer models to seek out people to help them with a problem, or to offer help. The positive effect of peer models was a surprise to some, as they had not expected to benefit from viewing peer models. However, a few felt that peer models had a negative effect on their learning either by reducing their confidence (weak students), or by making them complacent (strong students). The latter is perhaps less worrying, as proficient students may refocus their efforts towards other subjects that need attention. However, weaker students clearly need more work, but may become less motivated, believing that they are so far behind that they may never achieve the targets. This causes a problem for system designers - do we allow students to open their learner model to others as this can be beneficial to many, or do we protect the few for whom this may be demotivating?

Opening the learner model to peers (UMPTEEN)

- It creates an atmosphere of positive competition.
- If I open my learner model to my peers, they are more likely to let me view their models as well. Collaboration is important.
- I opened it to my friends because we can know us well and then we can help each other, which is good for improving our studies. I opened my model with personal details to xxx just because that he opened it to me for civility although I didn't plan to do that before.
- Several students doing this are top students who have excellent records, and don't need to hide their learner models.
- My learner model was not good. Therefore I did not open it. I thought I would feel a shame if I opened it. After all, I do not have concrete reason to be kind enough to do it, so I did not.
- As I didn't get a good mark in the test, I opened my learner model to all the peers anonymously. Lots of my classmates did better than me. Maybe this will make someone who did as bad as me feel better. At least, he or she was accompanied.
- The reason for making my learner model anonymous is the same reason why people like to get their results without other people knowing. If you have done poorly then you don't really want people to know that fact. In contrast if you obtain 100% then I wouldn't want people to know because you would have a stigma attached to your character.
- I asked for some people to whom I showed my model to open their models because I could compare against theirs. In addition there were some people asked me to open their model to them and they opened theirs to me at the same time.
- Whatever I did well or badly, it's just my business. And I do think there's no need to let others know, especially someone I don't know very well.

While some students opened their learner model to all or several peers, a second tendency was to open it to friends and sometimes a small set of selected others. Often students commented that they had opened their model to someone who was not a friend, simply because that person had opened their model to them, and opening their own model therefore seemed 'fair'. Some good students were amongst those opening their models widely, and some who had weak knowledge chose to keep it hidden. However, this was not always the case. The option to open one's model anonymously encouraged some to open a weak model to help other students with problems realise that they are not alone. Furthermore, some strong students opened their model anonymously in order to avoid people thinking less of them for having done well (for example, similar to the excerpt above, another student stated that they did not want to be thought of as 'showing off'). Some students organised with others to swap their learner models, and then did so, but only after making sure that both partners agreed to do this. Often the reason for opening the model to others was to promote collaboration and increase understanding, but sometimes also to encourage a competitive atmosphere which involved working individually towards improving one's own learner model. In contrast, some students did not wish to open their model, viewing learning as a private and individual matter.

As shown in the previous section, a majority of students (80%) did open their model to others - to varying numbers of peers. Those who do not wish to release their model should not have to (a matter of privacy). However, as most appear willing to do so, and most felt they benefited from viewing peer models, this approach looks promising. But the issue identified above, of a minority for whom this may have negative consequences, needs to be considered.

Opening the learner model to instructors (UMPTEEN)

- If the instructor just know my learner model without my name, it will only help the instructor to have a general view of all the students' learning process, thus they can't give me individualized tutoring. If instructors know more about my learning process, they will give me more correct guidance. So why not let them know the details?
- The reason I anonymously opened my learner model is to ensure that the instructors get the objective information of their students. It is the content of the learner model rather than the names of students that really matters.

- I didn't want to open my learner model to all instructors because my performance in the test is not good enough, even shameful. Plus the learner model has told me about which concept I am not good at and in which concept I should put my shoulder to the wheel.

Reactions to opening the model to instructors were more mixed. There was still a tendency to view this positively (even amongst some who did not release their model), either to provide information about problems of the group (which could be achieved with anonymous learner models), or to enable an instructor to give individual advice (which would require a student to open their model with their personal identifying details). However, some students did not wish to open their model even anonymously, because they felt ashamed that they did not know more. As students can use their learner model to identify their problems, some do not perceive a need to make it available to their instructor. As with opening the model to peers, students can be given the choice about whether to release their model to their instructor.

A simple open learner model for children (Wandies)

Because of their age, children, were not asked for written descriptions. The points below are based on experimenter observation and discussion with children and teachers.

- 7-8 year-old children of all skill levels understood the purpose of the OLM.
- Children were keen to practise their English in order to gain more gold wands.
- Children achieved a greater awareness of their educational needs.
- Children developed a team spirit.

Given that children tended to improve their performance in both conditions, Wandies could be used individually or in pairs. However, there was a greater individual increase amongst the paired students. Perhaps the most interesting effect, and not the original intention, was that children helped each other outside their pairings, if they saw that others had red wands. Such collaboration and assistance occurred also with university students using UMPTEEN, but in the case of Wandies it was as a result of a child opportunistically seeing an OLM on a screen (rather than intenionally seeking out learning partners as occurred with the university students). The group developed a team spirit, wanting everyone to succeed.

On an individual level, regardless of whether they were working individually or in pairs, half the children said that they had practiced English during their lunch break, in order to improve their learner model in the afternoon. This supports the view that an OLM can encourage children to learn independently, and to improve their knowledge. It had not been anticipated that children might work during their break. We do not know, therefore, whether the increase in performance was due to children's use of the system or due to their independent work. Nevertheless, the OLM was at least the prompt for this work. Of course, the motivation may have been to achieve gold wands rather than to learn, but learning was certainly a by-product of this activity. Two thirds of the children stated that they wanted to practice what they had learned at home, in order to make all of their magic wands gold.

The class teachers noted that after using Wandies, children referred to it in class when answering questions. Higher skilled children spontaneously referred to the OLM and how they had used it to improve their understanding of the different areas. Most children stated a desire to change all their magic wands to gold. The teachers stated that the system and the learner model in particular, seemed to give the children greater responsibility for learning, as the representation of understanding helped them to realise their educational needs. This is often an aim of OLMs for adults, but it seems that it can also apply with a simple OLM for quite young learners. In addition, the teachers felt that the children not only improved their approach to learning, but were also taking more interest in the understanding of classmates. Both teachers reported an increase in interaction between children, commenting that participants had broken the communication barrier that had existed between the sexes. It was observed that students were more frequently asking each other for help, and participants were more willing to share their work and help classmates, than other children in the class - even those who would not normally be expected to interact with each other. The teachers commented that a sense of camaraderie had built up amongst the group.

As stated previously, an aim of Wandies is to get young learners into the practice of reflecting on their knowledge with an OLM (while they are still eager), to ease the transition to more complex OLMs as they become older and more able to understand detailed or more structured representations of their understanding. If children have already used the coloured wands of Wandies, they may more easily use OLMs such as that of Zapata-Rivera & Greer [11], which externalises a Bayesian network using colour for 10-13 year olds; or the example in Figure 4 showing the views of the OLM in The Fractionator for 10-11 year old children. The Fractionator (using a superhero character called "Fractionator"), provides a choice of 4 methods of viewing knowledge of fractions, the options using colour in various formats: textual label of level of understanding with colour indicating the extent of evidence; colour representing knowledge level with size of circles indicating extent of topic covered; colour indicating knowledge level, also reflecting prerequisites structure; colour showing knowledge level, uncovering a picture as understanding increases. More complex examples for older learners include Flexi-OLM, an approach which could also be used at school level by older children. An investigation of The Fractionator with 10 users showed that, similar to Flexi-OLM with university students, all views received positive ratings, and children had different preferred views. Thus there are aspects of use of both Wandies and The Fractionator that are maintained with more complex OLMs used by adults.

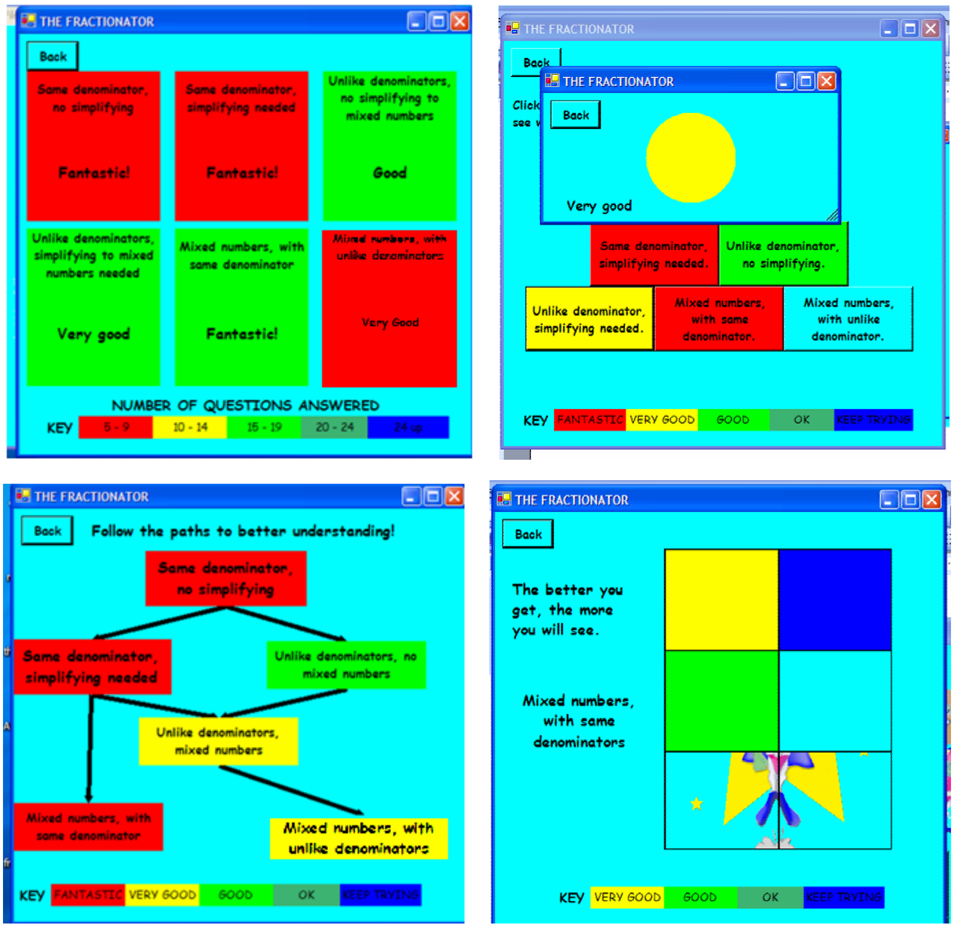


Fig. 4. The four views of The Fractionator OLM (10-11 year olds)

5. Summary and Conclusions

This paper has provided a descriptive account of students' reactions to open learner models. It has not attempted to offer vigorous experimental results - more formal studies will be available elsewhere. The focus here is on individuals - useful additional information for those considering opening the learner model of a system, if access to the model is intended as an important part of an interaction. If students do not use OLMs, we need to know more about why, in order to be able to judge whether it might be possible or useful to try to change their perception of them, or whether changes to the OLMs might render them more appealing. If users do use them, it is useful to know which features are liked and found helpful, in order to harness these aspects in our systems. Our examples have shown positive results for 7-8 year old children, which may be a useful way to accustomise young learners to the use of OLMs in order to ease the transition to the use of more complex presentations of their knowledge as they grow older. There were also positive results for OLMs for university students, and student comments showed that they clearly understood the purpose of an OLM, and could and would use them appropriately. Providing multiple views of the learner model seems useful. However, while the majority felt there to be benefit in using peer models, there is clearly a problem with a few individuals for whom peer models may be detrimental to their learning. Of course, a learner does not have to view the learner models of peers, but if these are available, it may be difficult for some students to ignore them. Do we continue with the approach of opening models to peers to benefit the majority, or do we withhold this functionality to protect a minority of students?

The next stage is to uncover the reactions of students making longer-term use of an OLM, for example, to support their learning throughout a term.

References

[1] Corbett, A.T. & Bhatnagar, A. (1997). Student Modeling in the ACT Programming Tutor: Adjusting a Procedural Learning Model with Declarative Knowledge, *User Modeling: Proceedings of 6th International Conference*, Springer Wien New York, 243-254.

[2] Weber, G. & Brusilovsky, P. (2001). ELM-ART: An Adaptive Versatile System for Web-Based Instruction, *International Journal of Artificial Intelligence in Education* 12(4), 351-384.

[3] Kay, J. (1997). Learner Know Thyself: Student Models to Give Learner Control and Responsibility, *Proceedings of International Conference on Computers in Education*, AACE, 17-24.

[4] Dimitrova, V. (2003). StyLE-OLM: Interactive Open Learner Modelling, *International Journal of Artificial Intelligence in Education* 13(1), 35-78.

[5] Bull, S. & Pain, H. (1995). 'Did I say what I think I said, and do you agree with me?': Inspecting and Questioning the Student Model, *Proceedings of World Conference on Artificial Intelligence in Education*, AACE, Charlottesville, VA, 1995, 501-508.

[6] Bull, S. & Kay, J. (2005). A Framework for Designing and Analysing Open Learner Modelling, to appear in Proceedings of *Workshop on Learner Modelling for Reflection*, AIED 2005.

[7] Bull, S. (2004). Supporting Learning with Open Learner Models, *Proceedings of 4th Hellenic Conference in Information and Communication Technologies in Education*, Athens, 47-61.

[8] Mabbott, A. & Bull, S. (2004). Alternative Views on Knowledge: Presentation of Open Learner Models, *Intelligent Tutoring Systems: 7th Int. Conf.*, Springer-Verlag, Berlin Heidelberg, 689-698.

[9] Bull, S., Abu-Issa, A.S., Ghag, H. & Lloyd, T. (2005). Some Unusual Open Learner Models, to appear in *Proceedings of International Conference on Artificial Intelligence in Education*.

[10] Bull, S. & McKay, M. (2004). An Open Learner Model for Children and Teachers: Inspecting Knowledge Level of Individuals and Peers, *Intelligent Tutoring Systems: 7th Int. Conf.*, Springer-Verlag, Berlin Heidelberg, 646-655.

[11] Zapata-Rivera, J.D. & Greer, J.E. (2002). Exploring Various Guidance Mechanisms to Support Interaction with Inspectable Learner Models, *Intelligent Tutoring Systems: 6th Int. Conf.,* Springer-Verlag, Berlin, Heidelberg, 442-452.

Web Framework for Scrutable Adaptation

Marek CZARKOWSKI, Judy KAY, Serena POTTS
School of Information Technologies
University of Sydney
Sydney, NSW 2006, Australia

Abstract
This paper describes a new system for delivery of personalized learning materials in a manner that enables the learner to scrutinize the adaptation, meaning that the learner can determine what has been personalized and what processes caused the particular form presented. We briefly explain the motivation for this new system, in terms of the difficulty of supporting such scrutability effectively as well as the intrinsic merit of the scrutability. We describe the user view of the system and report an evaluation of the way that learners interacted with it. We found that users were unaccustomed to the notion that they might understand and control personalization. Even so, they were able to scrutinize the adaptation in this new adaptive hypertext delivery environment.

1. Introduction

There are many reasons for personalizing web-based teaching materials, as reflected in the body of adaptive hypertext research. Typically, adaptive hypertext is presented to the learner without any indication of what has been personalized. So, for example, the learner may never be aware that they were given more quiz questions than a peer. Even if they do manage to discover this, or other forms of different adaptation, they typically have no way of working out why this is so. We believe that it is important to be able to build personalized systems in such a way that learners can scrutinize the adaptation of a hypertext system to answer these questions:

- What has been adapted to me?
- What caused the adaptation I saw compared with that seen by a peer?
- How can I control or alter the adaptation?

We have several motivations for this as argued in detail elsewhere [1, 5, 6]. Briefly we recap some of the reasons here. We want to be able to give learners control over their hypertext personalization. This is closely related to wanting learners to feel responsible for their own learning: such responsibility needs to be linked to control. We want to support learner reflection, based on seeing the learner model and the effects it has on the learning materials presented. All of these have been argued to help improve learning, especially deep learning. We also want to make the creators of adaptive hypertext accountable in the sense that their systems should be able to explain the basis for the personalisations performed.

In previous work [1, 5, 6], we have been quite surprised at the difficulty of providing an adaptive hypertext interface that learners are able to scrutinize to answer the questions above.

Part of the problem appears to be that people tend to be unaware of the fact that material has been personalized. Even if they realize this, they have difficulty appreciating that the personalization is driven by their student model. Even if they realize this, they have difficulty realizing that they can simply change their user model to effect changes in the personalization.
We do NOT believe that all adaptive hypertext needs to be scrutable. However, we believe it is important to be able to create scrutable adaptive hypertext delivery environments that enable learners to answer the above questions. Accordingly, we have designed a new adaptive hypertext interface so that we can explore a new approach to supporting scrutability. Essentially, it takes the rather radical approach of always presenting salient details of the student model along with the adapted content. It also provides single mouse action access to details of the adaptation and to the driver for it. We reasoned that this made it as easy as possible to answer the three questions.

Section 2 presents the user's view of the delivery interface. Section 3 describes the design of the evaluation, Section 4 reports the results and Section 5 draws conclusions.

2. System Overview

The system will be described based on a tutorial on UNIX file permissions. The underlying functionality can be used to build an adaptive learning system based on any subject. When logging in for the first time, the user is asked to answer a list of profile questions. They then see the coursemap page as shown in Figure 1. Subsequent logins show the user the coursemap page. Elements of the page are:

1. Coursemap cell – gives a list of the course sections, each with a link to the corresponding lesson page. The coursemap is adapted to the user based on their user profile. This is the left-most cell on the page.
2. Authentication cell – showing the user's login details and status.
3. Personalisation cell – has a link 'Change your Profile' that opens a new browser window with the interface to the user's profile interface, and the elements are summarised in this cell.
4. User Model cell – has a link 'View your user model' to the detailed user model.

Lesson pages contain the content about the section a user has selected. In the case of Figure 1, this is adapted on the basis of the aspects shown in the personalization cell. For example, the profile shows the user does not want examples but does want exercises. Each lesson page also has navigation arrows which link to the next and previous pages, to make the system easier to navigate. Lesson pages are customised in accordance with the user profile with some pages omitted if a user is not ready to view that page. Note the '*Personalisation'* and '*User Model*' cells are also present on every page, allowing the user to see the values of their user profile at any time.

If a page has been adapted, '*How was this page adapted to you'* is displayed at the end of the coursemap or lesson page. This can be seen in Figure 1. Alternatively to indicate that a page has not been adapted, '*This page has not been adapted*' is displayed at the end of the coursemap or lesson page. By default, the page displayed for a user has been adapted based on their user model. They cannot see what has been adapted or why. By clicking on '*How was this page adapted to you*?' at the end of a coursemap or lesson page, the same page will be displayed, this time highlighting sections that were included based on the user's user model in yellow and sections that were excluded in green. Figure 2 shows what happens when this is done for the screen shown in Figure 1. Once the adaptation can be seen, it can be removed by clicking '*Hide adaptatio*n' as seen at the lower left in Figure 2.

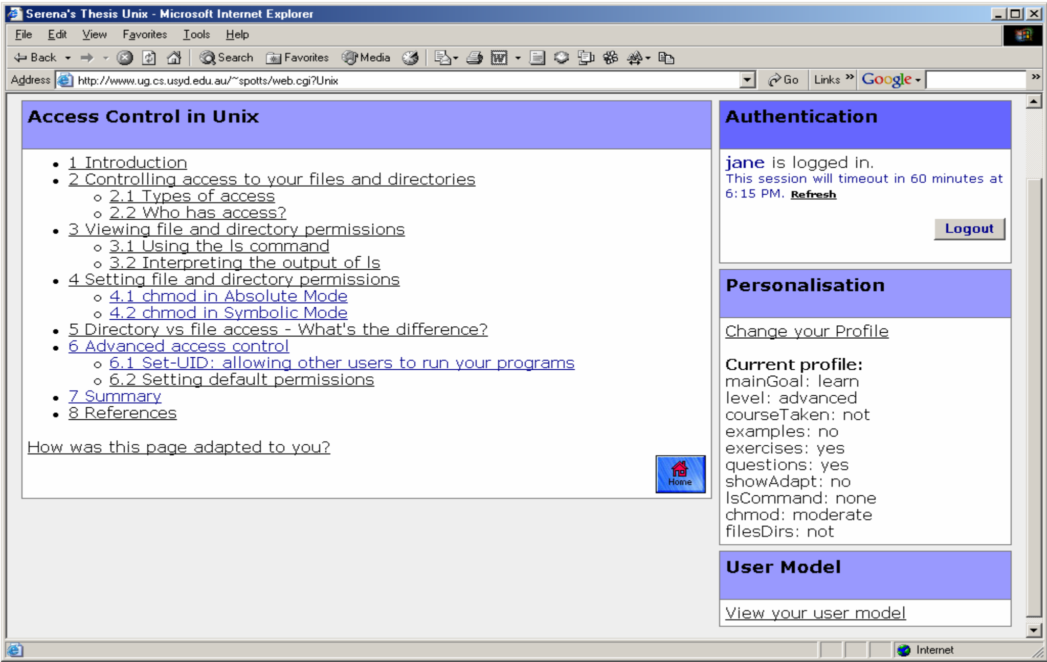


Figure 1: Coursemap page showing the four different cells contained on the page

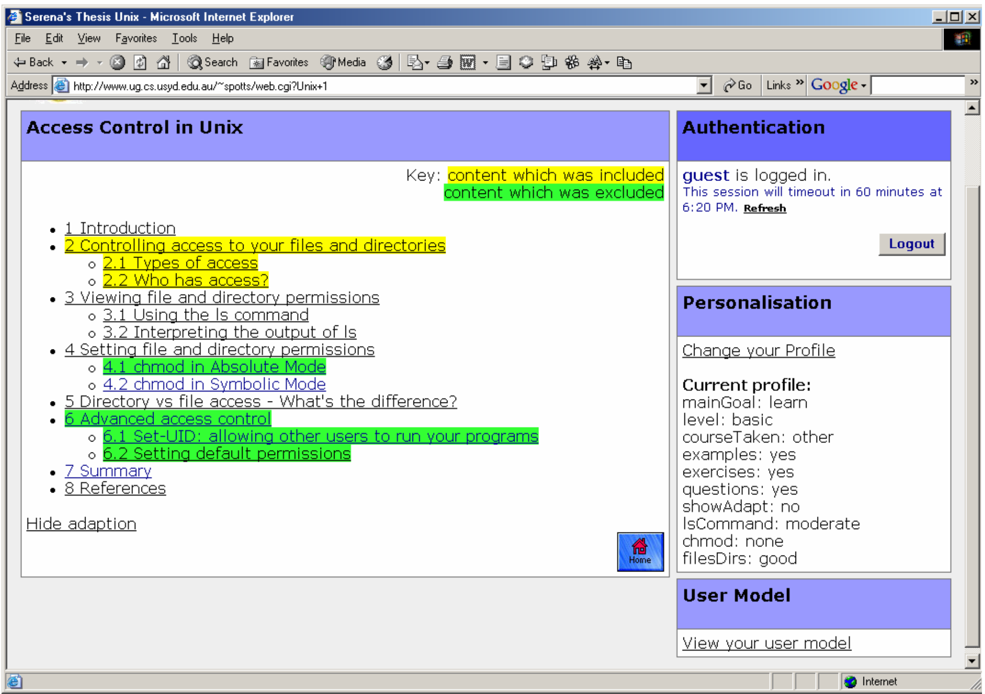


Figure 2: Coursemap that has been adapted according to the user's user model showing sections included highlighted in yellow and sections excluded in green

To see why an individual section has been included or excluded, the mouse is held over the section in question, and a caption will pop up to indicate the reason. Figure 3 shows an example of this use of the mouseover to show why content was included. This is indicated by the caption displayed: '*This was included because your level was: basic'*. Similarly, if the mouse was held over an excluded piece of text, a mouseover explaining the reason for the exclusion would be presented.

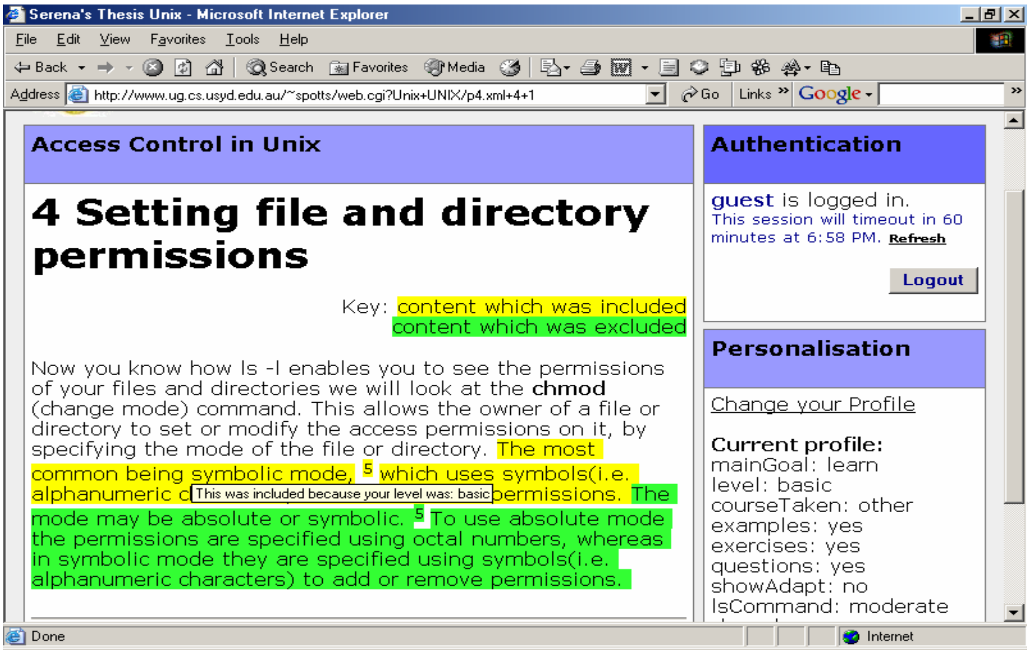


Figure 3: Mouseover showing content that was included indicated by the caption: 'This was included because your level was: basic'.

To change their profile, the user can always click '*Change your profile'* in the Personalisation cell. A page similar to Figure 4 will be displayed. The user is then able to change their profile. A message titled '*Your profile has been updated'* is displayed at the top of the '*Your Profile'* page to let the user know the update was successful. Pages within the system will be adapted based on this profile from now on.

The system has been built using a local lightweight but highly adaptable web framework called Cellerator [2]. Taking the categorisation by Bicking [3], it is a script-based framework. We have used it in conjunction with Personislite [7], which provides scrutable user modeling. Following the terminology of Brusilovsky [4], the system provides adaptive presentation and adaptive navigation. We have also drawn on many elements of the previous implementation of a scrutably adaptive hypertext [1, 5, 6].

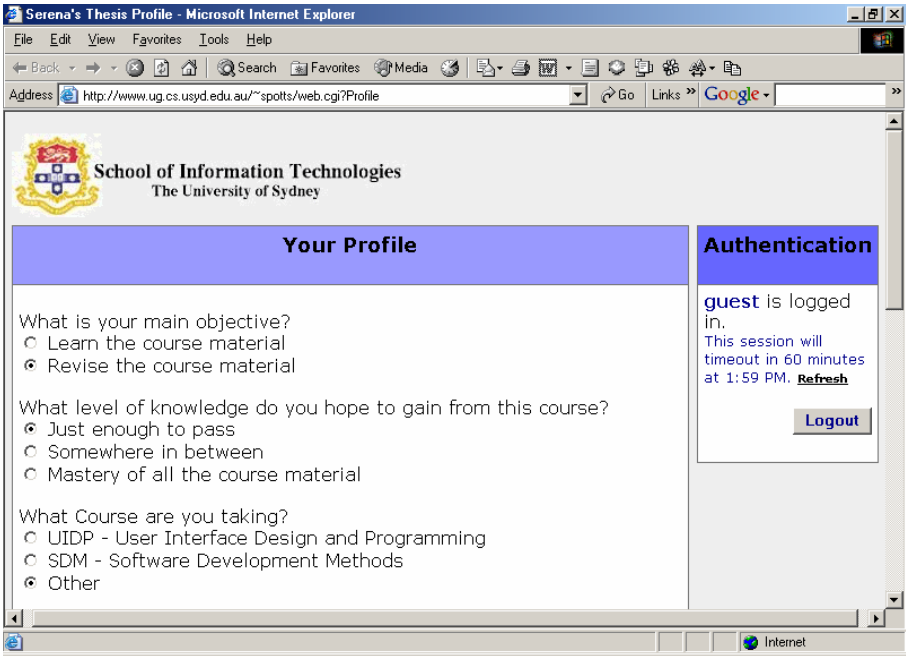


Figure 4: Change profile screen showing questions from the user profile before the user has changed their profile.

3. Evaluation

The system was used to assess whether participants could:

- Appreciate that their profile caused the adaptation;
- Determine what had been adapted to them;
- Understand why it had been adapted and
- Change their user model to control the adaptation.

We also wanted to explore affective issues, including whether particpants thought that the personalisation was helpful in a learning context and their attitudes to the interface.

The evaluation was undertaken in two stages. In first stage, participants 1-5 were asked to answer the initial questionnaire as if they were a single fictitious user, Fred, as in [1]. By contrast, participants 6-9, in the second stage, answered for themselves. In the first stage, all users had similar experiences and so we could frame tasks and questions accordingly. The second stage sacrificed this comparability and consistency but complements the evaluation by providing insights into the experiences of users who do not have the potential cognitive burden of remembering the profile elements for Fred. In our earlier work [1], we took the approach of the first stage and concluded that this may have biased the results. Accordingly, in this study, we explored both a consistent but inauthentic user model (Stage 1) and the authentic Stage 2 user model, where different participants, giving different answers to the user model questionnaire, would experience different adaptivity.

Participants were informed that our goal was to evaluate the system, not them. They were told it was an adaptive learning system based on UNIX file permissions and were asked to answer a series of questions about the system as they navigated their way through it.

Participants were instructed to work through *Section 1 Introduction*, which explained what they would learn in the course and gave a brief description of how the system works. They then answered questions which assessed how well they had understood how the system uses their profile settings and whether they would be able to influence or control the adaptation of the system.

Participants were then instructed to continue working through to the end of the tutorial, then answering questions relating to their understanding of how content is adapted by the system and why it was included or excluded based on their user profile. When they had completed the tutorial participants in Stage 1 answered questions which assessed whether they could predict how content would change if they altered their profile. We expected them to use mouseovers to see what content was included or excluded.

An error in the personalisation was purposely included in the system. We reasoned that this is just the sort of situation in which it would be natural for a user to want to scrutinise the personalisation to determine why the system appeared to behaving unexpectedly. We wanted to see if participants identified that the system had adapted something incorrectly based on their profile. The errors involved the presentation of exercises on every content page even though the answer to the profile question about trying relevant exercises was answered as ‘No’ initially.

4. Results

The results of the evaluation are summarised in Table 1.

Table 1: Summary of the user understanding

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Know about UNIX file permissions	No	No	Yes	No	No	Yes	No	A bit	Sort of
Understood role of Profile	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Understood coloured sections showed included/ excluded content	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Understood profile change alters adaptation	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Could identify excluded content	No	Partly	Partly	Yes	Yes	Yes	Yes	Yes	Partly
Used mouseover to see why content was included/ excluded	No	Partly	No	Yes	Yes	Yes	Yes	Yes	No
Could predict content change if their profile was changed.	No	No	No	Yes	Yes	n/a	n/a	n/a	n/a
Time to complete worksheet (minutes)	11	13	19	16	14	17	13	31	16

All first stage participants understood that their user profile would cause the adaptation of content within the system and were able to effectively change their user profile. Most were able to view the adaptation though of those who could, the majority experienced difficulties utilising the mouseover function provided to see the reason for the adaptation. This is a significant improvement over the previous study [1].

The second stage indicated that in general participants found the system easy to use and were able to use all the functions provided. Only Participant 9 appeared to experience trouble using the system. This appeared to be due to the fact they chose to see the adaptation on every page when they filled in their user profile initially. This may have confused them, as they did not know they could turn this feature off or that the system was able to display the content without the adaptation. Another user also chose to see all pages adapted and didn't have any problems using the system. So we cannot say conclusively this was the problem.

For the affective questions, all but Participant 8 in Stage 2 favoured the idea of adaptation. All gave positive feedback on the system design with its clear layout, good navigational tools and helpful instructions and explanations as appropriate.

The 'bug' in the system was not uncovered in either stage, even when participants were informed in Stage 2 that there was a bug. This may have been because the learners were very absorbed in the learning tasks: the initial questionnaire indicated that the participants did not know the content of the tutorial initially and all reported satisfaction in the quality of the learning experience. Participants in Stage 2 appeared to be more motivated than those in Stage 1, which may have contributed to their understanding of the system, as they were willing to learn and may have been paying more attention.

5. Discussion and Conclusions

The purpose of a scrutable adaptive hypertext system is to give the user control, allowing them to understand and control the adaptation. Our system provides the following features:

- A scrutable adaptive coursemap.
- Scrutable adaptive lesson pages.
- A modifiable user profile.
- A viewable user model.

The majority of participants:

- Could identify that their profile caused the adaptation;
- Were able to see what had been adapted to them;
- Understood why it had been adapted;
- Could change their user model, hence controlling the adaptation.

This is a real step forward, compared with our previous studies [1] with a more subtle interface. This study seems to indicate that if we want learners to scrutinize an adaptive hypertext learning environment, we may need the blatant always-present reminder that adaptation is being performed as we have done in the *Personalisation* cell at the right of the interface. It seems that users needed to be introduced to the idea of personalisation and then to scrutability, as our study indicates that for some it is not intuitive. We also conclude that since users need time to understand the notion of scrutable adaptation, it seems inappropriate to offer the initial, often poorly understood, option of setting all adaptations to be shown on all pages. Only after some use of the system should they be allowed to select this option.

The new system and its evaluation build upon our previous work to provide scrutably adaptive hypertext. Adaptive systems typically do not give the user the option to even see adaptation; this must contribute to the difficulty of supporting scrutability. With increasing concern for privacy, control over personal information and responsibility for one's own learning, it may be important to provide the option of scrutable interfaces such as we have described here. We still have much work to do in moving to more complex adaptation based upon larger learner models. We also still need to perform more extensive evaluations of this approach to include more opportunities for learners to have authentic needs to scrutinize the adaptation and to explore cases where that scrutinizing is more clearly linked to potentially improved learning outcomes.

6. References

[1] Czarkowski, M and J Kay, (to appear) Giving learners a real sense of control over adaptivity, even if they are not quite ready for it yet, Chen S and G Magoulas (eds), *Advances in Web-based Education: Personalized Learning Environments*, IDEA, to appear.

[2] Kummerfeld, B. and P. Lauder, *Cellerator,* http://www.it.usyd.edu.au/~bob, accessed August 2003

[3] Bicking, I., *The Web Framework Shootout,* http://www.python.org/pycon/papers/framework/web.html, accessed August 2003

[4] Brusilovsky, P., *Adaptive Hypermedia: From Intelligent Tutoring Systems to Web-Based Education.* Intelligent Tutoring Systems 2000, 2000: p. 1-7.

[5] Czarkowski, M., *An adaptive hypertext teaching system.* Honours Thesis, Basser Dept of Computer Science, University of Sydney, 1998.

[6] Czarkowski, M. and J. Kay, *Challenges of Scrutable Adaptivity.* Proceedings of AIED Conference, 11th International Conference on Artificial Intelligence in Education, IOS Press, 2003: p. 404 - 407.

[7] Kay, J., B. Kummerfeld, and P. Lauder, *Personis: A Server for User Models.* Adaptive Hypermedia and Adaptive Web-Based Systems, AH 2002 Proceedings, 2002: p. 203-212.

An environment helping teachers to track students’ competencies

Stéphanie Jean-Daubias*, Carole Eyssautier-Bavay**
** LIRIS - Université Claude Bernard - Lyon 1*
Bâtiment Nautibus, 8 bd Niels Bohr - 69622 Villeurbanne Cedex - France
*** CLIPS-IMAG - Université Joseph Fourier - Grenoble 1*
BP 53-38000 Grenoble Cedex 9 - France

ABSTRACT. Our research is about reuse and exploitation of learners’ profiles by the teacher. We also propose that other actors of the learning situation, learners and institutions, benefit from the exploitation of these profiles. In PERLEA project, we want from one hand to model the students’ tracking process, and on the other hand to propose an environment giving tools to help actors reusing and exploiting profiles of all disciplines, whatever their level.
KEYWORDS: teaching assistant, learner’s tracking, metacognition, interoperability.

1. Introduction

Our research is about reusing existing learners’ profiles by the teacher and exploiting these profiles by all actors of the learning situation. This theoretical research is conducted within PERLEA project and is applied through EPROFILEA environment, which proposes a set of tools allowing to reuse and exploit learners’ profiles for all disciplines and all levels.
There are many learners’ profiles in the education process, from primary school to university and adult continuing education. A great number of various profiles can exist for the same learner: there are partial photographs of the learner’s learning state, each taken at a specific moment. A learner’s profile can be defined as information concerning a learner or a group of learners, collected or deduced from one or several pedagogical activities, computerized or not. Information contained in the learner’s profile can concern his knowledge, abilities, conceptions or his behaviour. Learner’s profiles concern any subjects and any school levels; they can come from different sources (paper-pencil resulting for example from a classical test created by a teacher, or ILE (Intelligent Learning Environment) opening or not their learners’ models [11]), different persons can be at the origin of their creation. These profiles contained various types of information (knowledge, abilities, know-how or behaviour), in different forms (marks, assessments) and are represented in different manners in a textual, a digital, a graphical, a hierarchical form [5], or in a notional graph form [12] [3].

2. PERLEA project

ILEs designed in research laboratories are still weakly used in classrooms. Whenever they are used, it is often punctually; those ILEs concern a small part of the instructional program and teachers only use them for a limited number of sessions. In addition, when ILEs produce learners’ profiles, these profiles are not reused in competencies management process into the class, despite the interest they represent. This can be explained more particularly by the difficulty to get and to exploit different information coming from various sources: currently, no tool exists allowing teachers reusing and exploiting data from ILE externalising their learners' profiles.

From these observations, we propose through PERLEA project a way to improve the integration of ILEs in education by providing links between the use of ILEs and teachers' everyday practices. These links consist of tools helping teachers to merge profiles coming from teachers' practices and profiles produced by ILEs, in order to exploit them together in the classroom. This leads us to face the issue of interoperability in ILE. First some ILEs can delegate the exploitation of their profiles to EPROFILEA. We actually think that it is necessary to dissociate in ILE the diagnosis step from the use of learners' profile step; only the second step is treated in our project. Then EPROFILEA can redirect the learner to an ILE suited to the difficulties highlighted during his profile presentation.

In PERLEA project, we aim to model the pupils' tracking processes, in particular through the study of profiles use by different actors of the learning situation[1]. We make the hypothesis that an environment based on such a model answers the teachers' needs about individualization of teaching and is of great benefit to these actors. We think indeed that the exploitation of learners' profiles can improve learning, more particularly by proposing metacognitive activities and activities (pencil and paper or computerized) suited to learners' profiles. To complete our project successfully, we work regularly with teachers and pedagogical experts following a suited design method [7]. For example, this is the way we identify teachers' practice and needs concerning the use of profiles in their classroom. We will naturally carefully evaluate the pertinence of our model and the real benefits for each type of actors of the developed environment following evaluation methods suited to ILE [10].

3. EPROFILEA environment

PERLEA project leads to the implementation of an ILE: EPROFILEA (Exploitation of PROFILes by tEachers and leArners) [6]. On the one hand this environment aims at assisting the teacher in his profiles management, whatever the concerned domain, the level and the origin of the profiles (created by the teacher himself or by an ILE). On the other hand, EPROFILEA has to allow the exploitation of these profiles by the different actors of the learning situation. Actors who are most concerned by our environment are teachers who have different profiles for each of their pupils and want to reuse and exploit them. But it is also interesting to allow the learners themselves to view, manipulate and even negotiate their profile [1]. This could allow them to know what the teacher or the system knows or thinks about them [8], in order to become aware of the state of their knowledge, of their weak or strong points [2]. This process fits into a metacognitive approach [9]. In addition, several institutions, such as schools or higher authorities, are concerned by the exploitation of learners' profiles. Learners' profiles can also help families to track their child's learning and to establish a dialogue with the learner and the teacher.

EPROFILEA consists of two stages: first setting up the profiles compatible with the environment and then their exploitation (cf. Figure 1). Reusing profiles first requires the description of their structure. This description is done by the teacher in the structures building module, called Bâtisseur (for builder). This module makes operational the profiles description language defined in PERLEA project. This language must allow describing the existing profiles, whether they come from an ILE or they are pencil and paper profiles, whatever the type of information they contain. A first prototype of this module has been developed and is currently being tested. Filling the structures built in Bâtisseur (in order to make up learners' profiles), can be done in two different manners depending on whether

[1] This is one of the issues of Carole Eyssautier's PhD Thesis, began in 2003 in Arcade team, in CLIPS-IMAG.

data are coming from pencil and paper profiles or are learners' model coming from an ILE. In the case of pencil and paper profiles, EPROFILEA includes an assistant, PROSE (PROfileS keyboarded by the tEacher), helping the teacher to type data for each of his pupils according to the profiles structure defined within BÂTISSEUR. It requires showing the progression of data input by learner and by profile component. The first prototype ever developed is currently being tested. In the case of profiles coming from ILEs, EPROFILEA proposes profiles conversion modules (the "TOURBILLONS", for whirl): interfaces between the ILE and EPROFILEA. We are developing a module helping an expert teacher to build a new TOURBILLON adapted to the ILE whose profiles he wants to reuse.

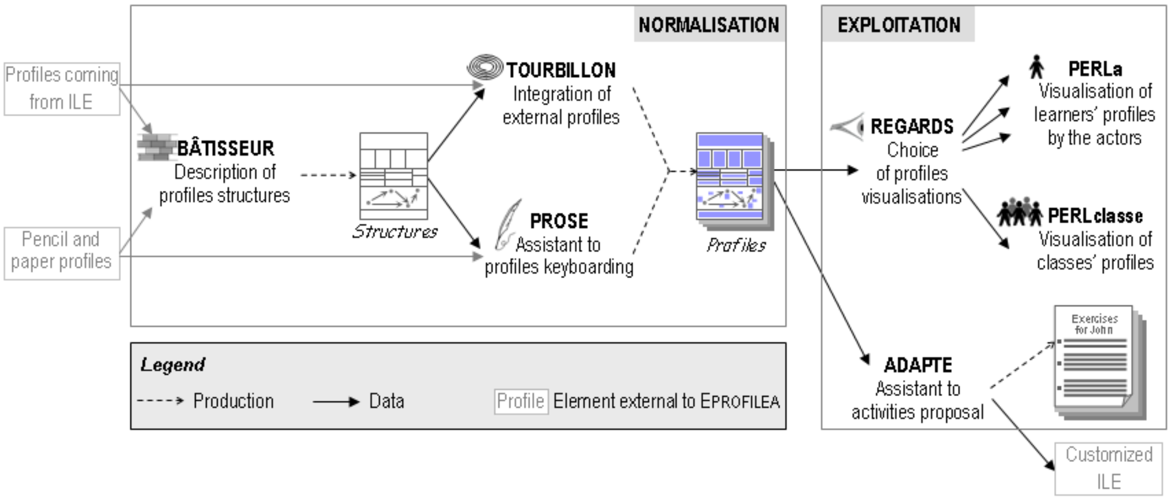


Figure 1: EPROFILEA architecture.

From the resulting profiles, the teacher establishes within REGARDS module (for views) the profiles visualisations that will be proposed to each actor of the learning situation. This module allows establishing different views for one profile: the teacher view, the learner view, the family view… To build these different views, the teacher chooses the parts of the profile that will be available for consultation by the different actors, the vocabulary used (suitable for these actors), and the representation mode (for example graphical, textual or numerical). PERL modules (Profiles of the lEarneRs expLoited) allow an interactive visualisation of the profiles by the different actors of the learning situation according to the view determined by the teacher in REGARDS module. There are several versions of these modules depending on whether the visualisation concerns learners' profiles or class' profiles, and depending on the actors: the teacher him-self, the learners, or even the institutions. We will propose different exploitations of learners' profiles allowing integrating in the best way information contained in the profiles in the learning process. The main exploitation consists of a visualization of the information adapted to the target public, supplemented by activities promoting their appropriation of the profiles. The design of these modules asks various questions: how to ease the appropriation of the profiles by the target public? How to represent the different profiles components corresponding to the different elements of the profiles description language? How to represent the profiles evolution over time? How to frame the profile negotiation between the learner and the teacher? In addition, we are planning to conceive and develop a last module, ADAPTE, allowing proposing learners activities suited to their competencies and suited to knowledge highlighted in their profiles: pencil and paper activities proposed by the system, or computerised activities managed by an other ILE. For this module, we must find a balance between genericity of EPROFILEA environment and disciplinary specificities or depending on the age, the scholar or academic level. We have to identify how it is first possible and then desirable to automate the activities creation. This module is from our point of view essential, insofar as it gives means to the teacher to include his work on the learners' profiles with EPROFILEA environment in his class practice into activities suited to the

profiles. If first simple prototypes of REGARDS and PERLe have been proposed and are currently been revised, ADAPTE is not treated for now.
Finally, there is a question transverse to all EPROFILEA modules: how to design, fill-in and exploit hybrid and progressive profiles. This includes information coming from different sources (pencil and paper or from ILEs), as well as the structure and the content that can evolve over time (such as addition of a part to the profiles frame by the teacher or taking into account the evolution of the learners' competencies during the school year).

4. Conclusion

In this paper, we presented PERLEA project and EPROFILEA environment which can be connected upstream with an ILE creating learners' profiles without allowing their exploitation by the actors and downstream with other ILE allowing a personalized course according to a specific profile. We have described EPROFILEA architecture helping actors to reuse and exploit learners' profiles of all disciplines, whatever their level, by teacher and different actors of the learning situation. This ILE is aimed at answering teachers' needs by proposing them a set of tools, support for profiles management and their reinvestment within classes' practices. We have already developed and tested some prototypes of these tools, we have in particular proposed an experimental device based on EPROFILEA architecture from description of the profiles up to their negotiation by learners [4].

5. References

[1] Brna, P., Self, J., Bull, S., Pain, H., Negociated Collaborative Assessment through Collaborative Student Modelling, Workshop Open, Interactive, and Other Overt Approaches to Learner Modelling, AIED'99, 1999, pp. 35-42.

[2] Bull, S., Nghiem, T., Helping Learners to Understand Themselves with a Learner Model Open to Students, Peers and instructors, Workshop Individual and Group Modelling Methods that help Learners Understand Themselves, ITS'02, 2002, pp. 5-13.

[3] Drimitova, V., Brna, P., Self, J., The design and implementation of a graphical communication medium for interactive open learner modelling, ITS'02, 2002, pp. 432-441.

[4] Eyssautier, C., Jean-Daubias, S., A Device Helping Learners to Self-Assess Themselves, CALIE'04, 2004, pp. 185-190.

[5] Hartley, D., Mitrovic, A., Supporting learning by opening the student model, ITS'02, 2002, pp. 453-462.

[6] Jean-Daubias, S., Exploitation de profils d'apprenant, EIAH'03, 2003, pp. 535-538.

[7] Jean-Daubias, S., De l'intégration de chercheurs, d'experts, d'enseignants et d'apprenants à la conception d'EIAH, TICE'04, 2004, pp. 290-297.

[8] Kay, J., Ontologies for reusable and scrutable student models, Workshop Ontologies for Intelligent Educational Systems, AIED'99, 1999.

[9] Noël, B., La métacognition, De Boeck Université, 1997.

[10] Nogry, S., Jean-Daubias, S., Ollagnier-Beldame M., Évaluation des EIAH : une nécessaire diversité des methods, TICE'04, 2004, pp. 265-271.

[11] Paiva, A., Self, J., Hartley, R., Externalising Learner Models, AIED'95, 1995, pp. 509-516.

[12] Zapata-Rivera, J.D., Greer, J., Externalising Learner Modelling Representations, External Representations in AIED: Multiple Forms and Multiple Roles Workshop, AIED'01, 2001, pp. 1-6.

Advanced Reporting Systems in Assessment Environments

Diego ZAPATA-RIVERA, Jody S. UNDERWOOD and Malcolm BAUER
Educational Testing Service,
Rosedale Road. Princeton, NJ 08541 USA

Abstract. Advances in measurement technology, cognitive science, and assessment design make it possible for assessment environments to provide rich and useful performance information to students, teachers, parents and administrators. This paper reports on how different technologies have been used to create innovative reports and describes our vision of advanced active reporting systems that do not only present information in various ways, but can also listen to and support formative dialogue among students, teachers, and parents. Thus, active reports can be instrumental in linking summative and formative assessments in service of student learning.

Introduction

Higher standards for accountability in US education as defined in the No Child Left Behind Act of 2001 (NCLB) have imposed new challenges on schools that now strive for getting reliable and valid information to assess and guide student learning. Educational assessment programs are beginning to produce score reports that provide valuable information about student performance at the student, school, district and state levels. Educational Testing Service (ETS) is developing innovative score reports that convey clear and useful diagnostic information using various kinds of representations that can be adapted to individual user needs.

Score reporting is one of the most important parts of assessment design because score reports is what test-takers and other score recipients use for decision-making. It is also the part to which assessment designers have traditionally paid relatively little attention. The problem is that scores alone do not give a lot of information about performance. If the score report cannot communicate this valued information well, the test cannot achieve its intended purpose.

Recent efforts in enhanced score report design, both in academia and in industry, use visuals and text descriptions to make scores more understandable and to divide performance reports into sections, subscores, and less often, skills. There has been some work in creating more descriptive reports, for example, making a link between assessment performance and state or national standards, presenting feedback that will help students understand their areas of strengths and weaknesses, and suggesting ways to improve in those areas.

Information to populate such diagnostic reports can be gathered and maintained in the form of cognitive diagnosis models. These models are characterized by providing information based on reliable and valid evidence gathered during the assessment. Score reports can be seen as views of such cognitive models.

Score reports typically provide a static view of a subset of a student's performance. Such reports can be presented on a piece of paper (i.e., paper-based score reports) or take the form of an interactive tool in which the user chooses the kind of information, level of detail, and external representation they want to see (e.g., web-based score reports). However, it is also possible to use these reports as a mechanism to capture what students, teachers, and parents think about the assessment and also as a communication tool that will enhance formative dialogues based on the results of the assessment. This information can be incorporated into the student model. We call these *active reports*.

Active reports have been proposed based on research done in the area of open/inspectable student models (ISMs). ISMs have been used to support student reflection, knowledge awareness, self and collaborative assessment, group formation, student model accuracy and learning [1, 2, 3, 6, 7, 9, 10, 11, 15, and 16]. Active reports can be used to support a continuous assessment process that will guide student learning long after an initial assessment is over.

This paper is structured as follows: Section 1 presents the PSAT/NMSQT® as an example of an assessment program that uses a cognitive model to produce a basic diagnostic score report. Section 2 presents assessment and score report design methodologies (Evidence-Centered Design (ECD), and User-Centered Design) that can be used to produce enhanced diagnostic score reports. Finally, drawing upon these technological advances and building upon previous research on open/inspectable Bayesian student models, we describe active reports (Section 3) in which score reports are enhanced in order to serve as communication tools and as mechanisms to acquire additional evidence of student knowledge and performance.

1. PSAT/NMSQT® Example

The Preliminary Scholastic Aptitude Test/National Merit Scholarship Qualifying Test (PSAT/NMSQT) individual score report exemplifies the kind of information available to students in some assessment programs (See Figure 1). PSAT/NMSQT provides students with their scores, score ranges and percentiles in three main sections: critical reading, math, and writing; the correct answer and difficulty level of the test questions; the student's answers; and for the math section whether the question content involved arithmetic, algebra, or geometry. In addition, individual score reports include skills feedback based on the student's performance on the test (i.e., skills that need improvement and general advice on how to improve them).

Information for this report is produced using a cognitive model for diagnosis based on Rule-Space [12]. Rule-Space uses a data structure called the Q-Matrix that represents a mapping from the test items to the skills that are required to solve them. Assessment specialists identify the targeted skills and fill out the Q-Matrix structure.

Rule Space assumes that the examinee tends to get an item right if he or she has mastered all the skills identified by the Q matrix. Using this model, posterior probabilities of skill mastery are computed based on the student's pattern of question responses. Finally, students are classified as master of the skill, as non-master (i.e., requires improvement on that skill), or as unknown (i.e., unable to tell) according to their posterior probability values.

Up to three non-mastered skills are selected to provide additional feedback. This feedback includes general comments for improvement and pointers to the questions in their test book for which each of these skills is mapped.

By linking test items to skills and including information about skills in the student score report, PSAT/NMSQT emphasizes skill improvement rather than right and wrong answers. This type of feedback can be used by students, teachers, and parents to enhance the learning process.

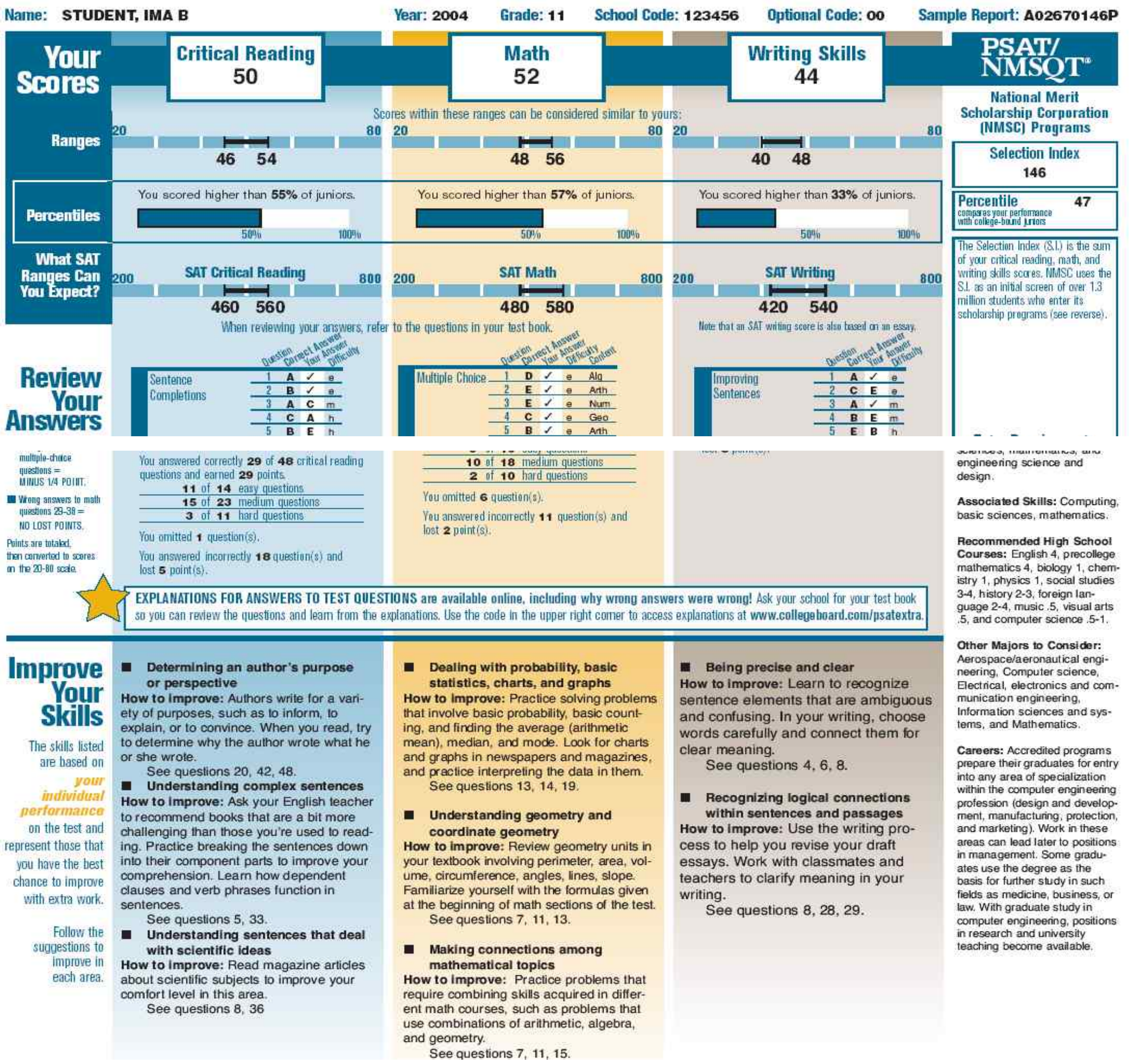


Figure 1. Fragment of a PSAT/NMSQT® score report

2. Assessment and Score Report Design Methodologies

2.1 Evidence-Centered Design

Cognitive analysis required to produce a model that provides reliable and valid information to generate enhanced reports is a complex task. Reliability and validity of the model used for the assessment depends on the quality of the evidence collected and on how it is used to measure student performance. For example, a student could have had a bad day for a particular test, or the test itself might not be designed well.

Evidence-Centered Design (ECD) [8] is a methodology employed at Educational Testing Service that emphasizes a logical and explicit representation of an evidence-based chain of reasoning for assessment design, which helps to keep the assessment valid and reliable. This evidence-based chain of reasoning is maintained through the production of proficiency, evidence, and task models. Proficiency models are used to define the knowledge, skills, abilities or traits that should be assessed and their interrelationships. Evidence models define how observations of behavior are considered as evidence of proficiencies. And, task models describe how assessment tasks must be structured to ensure that behaviors constituting evidence are observed. Figure 2 depicts three central models of ECD.

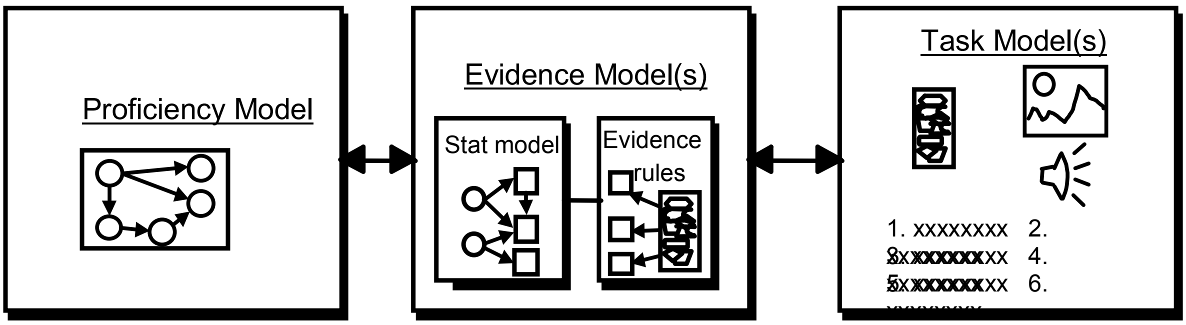


Figure 2. ECD central models

ECD is not limited to a particular cognitive domain model, type of task, type of evidence, model representation scheme, or scoring model. Instead, ECD provides general principles and tools to guide and support the assessment design process. Models can take on different forms. For example, proficiency and evidence models could be represented using Item Response Theory (IRT), Bayesian networks, number right, etc., while task models could take the form of multiple choice items, constructed response items, simulations, etc.

It is important to notice that the proficiency model has been separated from evidence and task models. This allows for choosing different evidence models according to available evidence and propagating it to the corresponding proficiencies in the proficiency model. A student model in this framework can be defined as an overlay on a proficiency model plus relevant supporting information drawn from evidence and task models.

2.2 User-Centered Design

The ETS Evolve project designed enhanced score report components for individuals and teachers that meet the needs of many programs and products. To develop the reports, researchers analyzed reporting requirements for a sample of state assessment Requests for Proposals and the U.S. federal No Child Left Behind act (NCLB), current practices on score report components (e.g., [5]), reviewed existing literature on information visualization (e.g., [13 and 14]), and collected data from parent and teacher focus groups representing elementary and secondary grades. These designs enhance reports by providing useful information, improving score report readability, and improving the use of assessments by educational stakeholders.

General characteristics of Evolve's enhanced score reports:

- Three different representations of information are presented in each reporting component to make the data accessible to different types of users. These include visuals, numbers, in addition to text descriptions.
- Each report component answers a key question (e.g., "What skills or knowledge does my child's performance reflect?" or "How did my child perform in comparison to other students in the school?"). Too much information, especially about varied questions, presented in one report component makes it difficult to understand
- Legends need to be the first thing that people see. When legends appear at the bottom of the page, people do not notice them.
- Displays are clear, consistent, and readable.
- Color scheme highlights data, in addition to sections of the report
- Reports are both computer-delivered and paper-delivered.
- Enhancements are geared toward web use, such as the ability to link to related information and rollovers that display definitions.
- Report representations and organization are tailored to individual client needs.

An example of an Evolve report component is shown in Figure 3. This component answers a particular question ("What are my child's overall scores in Math and Reading?"), personalizes the report by using the student's name, uses a graded color scheme that is easy to remember once you read the legend, and shows both numbers and a visual representation of the data. In the web version, it also shows a textual description of the student's performance and gives on-demand definitions and information.

Work is continuing in this area to develop easy-to-understand report components, as well as in the layout of the components for a complete report. It is in ETS' mission to present more and useful information in order to improve the use of assessments by educational stakeholders.

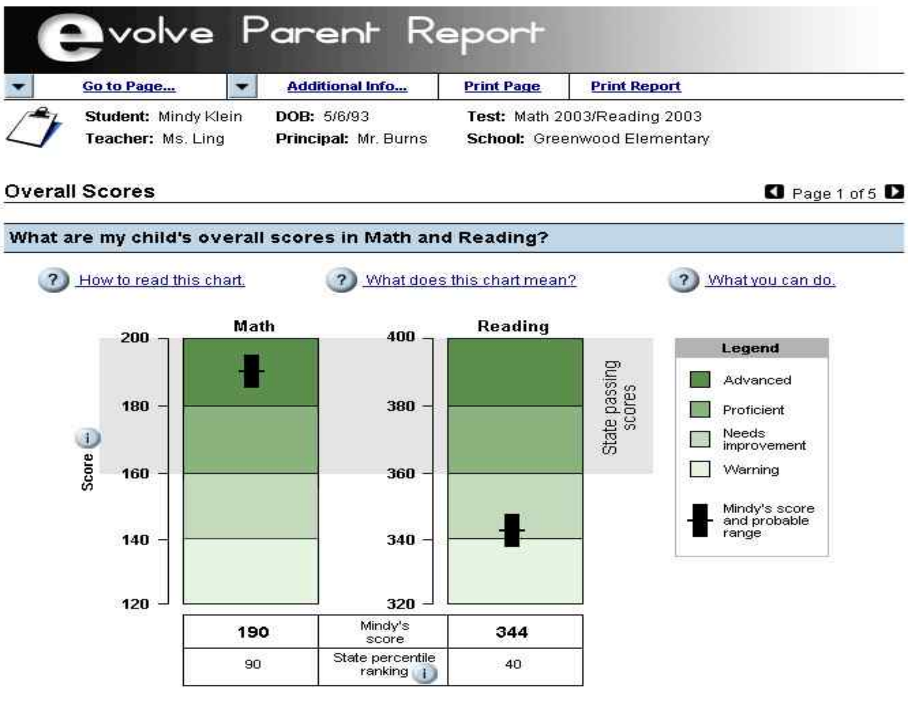


Figure 3. Evolve Parent Report

Score reports that address focused questions not only help people understand their intended message, but also help elicit interesting questions (e.g. "What can I do to help my child?"). Thus, appropriate score reports can be used to initiate dialogues that will benefit students, parents and teachers.

Information available in the form of cognitive models developed using ECD methodology plus enhanced diagnostic reports using user-centered design principles provide the basis for developing *active reports* that will function as communication tools and as mechanisms to gather additional evidence of student performance.

3. Active Reports

Active reports result from the idea of thinking about score reports as active entities that can be used to convey useful information, support communication among students, teachers, and parents and at the same time as a mechanism to acquire additional evidence of student knowledge and performance.

An active report is more than an interactive or adaptive report. It listens to students, teachers, and parents and uses this information to update the student model. It can present different views of the student model (e.g., an active report view for the teacher), as well as different aspects of the student model (e.g., different variables, different levels of granularity, different external representations).

Previous research in the area of open/inspectable Bayesian student models [15, 16, and 17] showed that students' and systems'/teachers' opinions could be modelled as unreliable sources of evidence in a multi-layered Bayesian structure. Different views of the Bayesian student model were made available to students and teachers (i.e. the teacher's view, the student's view, and an aggregated/negotiated view). Various guiding mechanisms were used to explore human interaction with Bayesian student models (i.e. solving a guiding questionnaire, explaining the model to a human peer, interacting with an artificial guiding agent, negotiating the status of the model with the teacher, and exploring the model as a group). Some of these guiding mechanisms directed students to particular areas of the model that presented conflicting evidence or that needed further attention. Interaction with these tools has shown the presence of student reflection, which also contributes towards learning.

Bayesian models can be used to integrate evidence of student knowledge and skills coming from summative and formative assessments. Teachers and students can use open/inspectable Bayesian student models to add and share assessment information (e.g. to refine a view of the student model by adding their opinions). Reports built out of these models could be designed to present a single view of the model (e.g. the teacher's view) or a composite view of the student model (e.g. a view that includes information from summative and formative assessments). In fact, being able to tell the whole story facilitates the use of assessment information for learning and decision making based on assessment information.

ECD models hold valuable student assessment information that can be made available to students, teachers, and parents through the use of active reports. Using active reports, a view of the ECD models containing information about student knowledge and supporting evidence can be presented to these stakeholders, who will interact with it and provide additional information gathered during the learning process (e.g. student self-assessment and results from formative assessments).

ECD model information linked to relevant educational materials can be used with active reports to act as computer-based learning environments. Evidence of student knowledge gathered using active reports could also be used to inform the ECD models. Thus, listening to students, teachers, and parents will not only enhance the ECD models, but will also support a formative learning process based on a view of the model that changes in light of new information.

Figure 4 shows how evidence gathered through active reports is modelled as a source of evidence in the ECD framework. After the student has taken a test, an active report is generated and presented to the student, teacher, and parents.

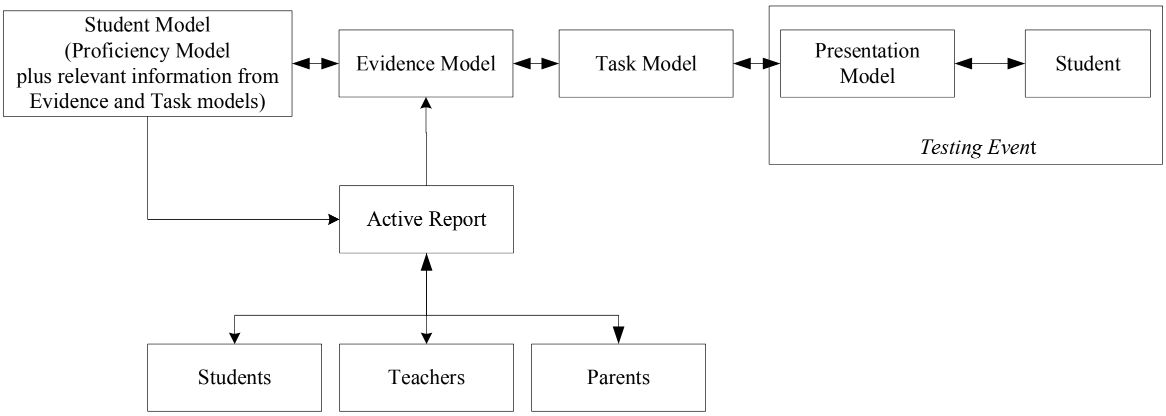


Figure 4. Active reports are used to present evidence of student knowledge/performance to and gather additional evidence from students, teachers and parents

Interacting with the student model through active reports can serve different purposes. For example, learning can be directed to unknown or weak topics, student reflection can be supported by providing appropriate guiding mechanisms, and inspection of the student model by teachers and parents can be facilitated, which can be important in mediating interactions between teachers and parents and in providing appropriate and timely feedback to students. Thus, as a communication tool, active reports can support synchronous and asynchronous formative communication among students, teachers, and parents based on the model. Results from an initial assessment can be used to power a learning process based on continuous-, self-, collaborative-, and negotiated-assessments [1].

Some scenarios in which active reports can be used are as follows:

- *Improving testing scores*. Students who want to improve their test scores or want to use a test to measure their knowledge/performance level in a domain area can use an active report to get access to a view of the student model that shows observed levels of knowledge/mastery. Students can use the active report to interact with educational resources linked to each of the student model variables that appear on the report. Additional evidence of student knowledge gathered during this practice phase will be used to update the student model, and students can get an estimate of how their performance has changed through time.

- *Supporting learning in a simulation-based assessment environment*. Students using a simulation-based assessment environment can interact with an active report that describes their performance. Using the active report as a guiding mechanism, students can be directed to particular areas of the simulation environment to practice and get additional information that can help them improve their performance. As students practice, the active report will gather evidence and will update the student model as in the previous scenario.
- *Sharing student model information in an e-learning environment*. By adding relevant educational resources in the form of learning objects, active reports can be used as e-learning environments in which as students work on a variety of learning activities, teachers and parents can witness the progress students have made. In addition, teachers can intervene to provide advice based on the information presented as part of the active report that is continuously updated and readily available.

In general, active reports can make assessment results relevant by supporting learning in the classroom. By actively supporting learning using assessment, active reports can help close the loop from summative to formative assessment in service of student learning.

4. Conclusions

Accountability reforms in US education have brought new demands for relevant and useful information of student performance. Assessment companies are beginning to provide innovative score reports to satisfy the needs of all educational stakeholders.

We have shown how ECD is being used to facilitate the creation of valid and reliable cognitive models and how principles of user-centered design have been used to improve score reports. Building upon these methodologies and research in the area of ISMs, we presented active reports as enhanced diagnostic score reports that not only present assessment information, but can also listen to and support formative dialogue among students, teachers, and parents based on the student model. The value of a report is not in the report itself, but in the thinking and communication that the report fosters. Active reports provide a means to support this thinking and communication at the right time - when people are using the report itself.

Current research efforts have been focussed on designing and implementing the technological infrastructure needed to produce and integrate active reports with current reporting and assessment design systems. Next steps include designing an initial prototype of an active report, and testing it with teachers, students and parents.

Advances in educational assessment and reporting taking place at ETS make it possible to think that research on open/inspectable student models in the form of active reports could soon be part of real products and services that positively affect the lives of students, teachers, and parents.

References

[1] Brna, P., Self, J., Bull, S., & Pain, H. (1999). Negotiated Collaborative Assessment through Collaborative Student Modelling. Proceedings of the workshop 'Open, Interactive, and other Overt Approaches to Learner Modelling' at AIED'99. Le Mans, France, pp. 35-44.

[2] Bull, S. & McKay, M. (2004). An Open Learner Model for Children and Teachers: Inspecting Knowledge Level of Individuals and Peers, in J.C. Lester, R.M. Vicari & F. Paraguacu (eds), Intelligent Tutoring Systems: 7th International Conference, Springer-Verlag, Berlin Heidelberg, 646-655.

[3] Dimitrova, V. (2003) STyLE-OLM: interactive open learner modelling. International Journal of Artificial Intelligence in Education, vol. 13, pp. 37-58.

[4] Gitomer, D., & Bennett, R.E. (2002) Unmasking Constructs Through New Technology, Measurement Theory, and Cognitive Science. In Technology and Assessment: Thinking Ahead Workshop. 1-11. Available on-line: http://www.nap.edu/openbook/0309083206/html/1.html

[5] Goodman, D. P. & Hambleton R.K. (2004) Student Test Score Reports and Interpretive Guides: Review of Current Practices and Suggestions for Future Research. Applied Measurement in Education, 17(2), 145–220.

[6] Hartley, D. & Mitrovic, A. (2002) Supporting Learning by opening the Student Model. In proceedings of ITS
2002. pp. 453-462.

[7] Kay, J. (1998). A Scrutable User Modelling Shell for User-Adapted Interaction. Ph.D. Thesis, Basser Department of Computer Science, University of Sydney, Sydney, Australia.

[8] Mislevy, R.J., Steinberg, L.S., & Almond, R.G. (2003). On the structure of educational assessments. Measurement: Interdisciplinary Research and Discussion. 1 (1) 3-62.

[9] Mitrovic, A., Brent, M. (2002) Evaluating the Effects of Open Student Models on Learning. AH 2002: 296-305

[10] Morales, R., Pain, H., & Conlon, T. (2001). Effects of Inspecting Learner Models on Learners' Abilities. Proceedings of AIED'01. IOS Press, 434-445.

[11] Paiva, A., Self. J. & Hartley, R. (1995). Externalising learner models. Proceedings of World Conference on Artificial Intelligence in Education, Washington DC, 509-516.

[12] Tatsuoka, K. K. (1995). Architecture of knowledge structures and cognitive diagnosis: A statistical pattern recognition and classification approach. In P. Nichols, R. Brennan, & S. F. Chipman (Eds.), Cognitively diagnostic assessment (pp. 327–359). Hillsdale, NJ: Erlbaum.

[13] Tufte, E.R. (1983). The visual display of quantitative information. Cheshire, CT: Graphics Press.

[14] Wainer, H., Hambleton, R.K., & Meara, K. (1999). Alternative displays for communicating NAEP results: A redesign and validity study. Journal of Educational Measurement, 36(4), 301-335.

[15] Zapata-Rivera, J.D. (2003) Learning Environments based on Inspectable Student Models. Ph.D. Thesis. University of Saskatchewan. Department of Computer Science.

[16] Zapata-Rivera, J.D., & Greer, J. (2004) Interacting with Bayesian Student Models. International Journal of Artificial Intelligence in Education. Vol. 14, Nr. 2 127-163.

[17] Zapata-Rivera, J.D., & Greer, J. (2004). Inspectable Bayesian Student Modelling Servers in Multi-Agent Tutoring Systems. International Journal of Human Computer Studies. Volume 61, Issue 4. 535-563. [Online] Available: http://dx.doi.org/10.1016/j.ijhcs.2003.12.017

Assess: Promoting Learner Reflection in Student Self-Assessment

Lichao LI, Judy KAY
School of Information Technologies,
The University of Sydney, NSW 2006, Australia
{lli1, judy}@it.usyd.edu.au

Abstract. Learner reflection is critical to effective, deep, transferable learning, especially in cognitively demanding areas, such as learning programming. This paper presents Assess, a programming education system that facilitates student self-evaluation and learner reflection. Our evaluations show that Assess helps students identify strengths and weaknesses and that they appreciate the learning value of the reflective interface available in Assess.

Keywords: teaching programming, learner reflection, student modeling

1 Introduction

Learner reflection is "a generic term for those intellectual and effective activities in which individuals engage to explore their experiences in order to lead to a new understanding and appreciation" [1] and it is most likely to occur when the learner encounters some problematic aspect of learning and attempts to make sense of it. There is a large body of evidence suggesting that learning effectiveness can be enhanced when learners pay attention to their own learning experiences by reflecting on the state of their knowledge and the learning process [2]. Learner reflection is especially important for cognitively demanding learning topics, such as learning programming. Schön identified three types of learner reflection, namely *reflection in action*, *reflection on action* and *reflection on reflection* [3], that may occur in any learning experience. We explore ways to support the first two of these in a conventional programming teaching/learning tool named Assess.

The work reported in this paper was conducted in the context of teaching/learning programming in C and Java in our undergraduate subject, Software Development Methods I, which is a second year programming course that aims to teach programming in C in a UNIX environment. The course starts with a brief revision of basic programming techniques from our first year Java courses. It then covers concepts such as pointers, memory management, dynamic data structures (linked lists in particular), useful standard library functions, basic UNIX commands and UNIX concepts (i.e. file system and redirection and pipes).

Assess is one of the learning support tools designed to help students improve their learning and helping them pass the course. The system has evolved considerably since it was first built in 2002. Its original objectives were to enable students to self-evaluate their knowledge, monitor their learning progress and judge if they have achieved the required learning outcomes. This paper describes a new version of Asses with a knowledge layer. Our research goal is to explore ways to add a knowledge layer with student modeling and support for improved feedback on learner's progress.

The following section gives an overview of Assess. Section 3 describes evaluations and Section 4 concludes with further work and conclusions.

2 System Description

To accomplish our research goal in the context of teaching programming in C and Java, domain knowledge of the subject was analyzed and teaching/learning goals as well as students' common misconceptions [4] were identified and summarised. Based on this information, a pragmatic learner model [5] was defined. This was used to create a model for each student's knowledge state and learning progress. A student's interactions with the system are also recorded in her learner model so that the model always holds the system's current beliefs about the student's knowledge state. Making this information available to students by scrutinising their user models in Scrutable Inference Viewer (SIV) [6] can help them become aware of their learning progress, thereby promoting learner reflection [7].

Assess has a collection of tasks, like the example in Figure 1. As the figure shows, there are three steps in the student self-evaluation process:

1. Students provide their own solutions to the problem;
2. They self-assess that solution using criteria the teacher has defined for that task and;
3. They read and assess example solutions provided by the teacher.

The problem statement page starts with a statement of what the teacher intends students should learn from the task. When writing an answer to the problem, students need to pay special attention to these learning objectives, as they are the particular concepts that the task aims to teach. For example, in Figure 1, the task is intended to help students learn about

1. Loops;
2. Pure Pointers[1] and;
3. Use of standard streams.

Once the student submits a solution, she can self-assess it against a set of marking schemes as in Figure 2. Each marking criterion asks the students to rate one aspect/component of her solution. For example, it may asks her to rate how well she has written a loop. She can choose from a range of options for each marking criteria, for example, excellent, good, ok, poor or rotten. This is where reflection-in-action occurs, as when students self-evaluate their solutions to a problem, they will read a marking criterion and reflect, as they rate their solution with respect to the criterion. They may also realise that they missed something in their solutions and since the criteria are shown in a new pop-up window, they can fix their solutions while evaluating them. For instance, when a student sees the last marking criteria in Figure 2, she may realize that she had forgotten to validate the input values. The system ensures that these criteria are consistent with the teacher's goals for the task by associating the teaching goals with the criteria. Thus, when a student is assessing his/her solution with the set of marking criteria, she will be able to reflect and learn the explicit concepts that are being developed in the task. At this stage, we need to use human tutors to grade the student solutions. However we also provide some code reading and reflection in the third stage.

[1] This is a term used in the subject, familiar to the students, referring to simple uses of C pointers.

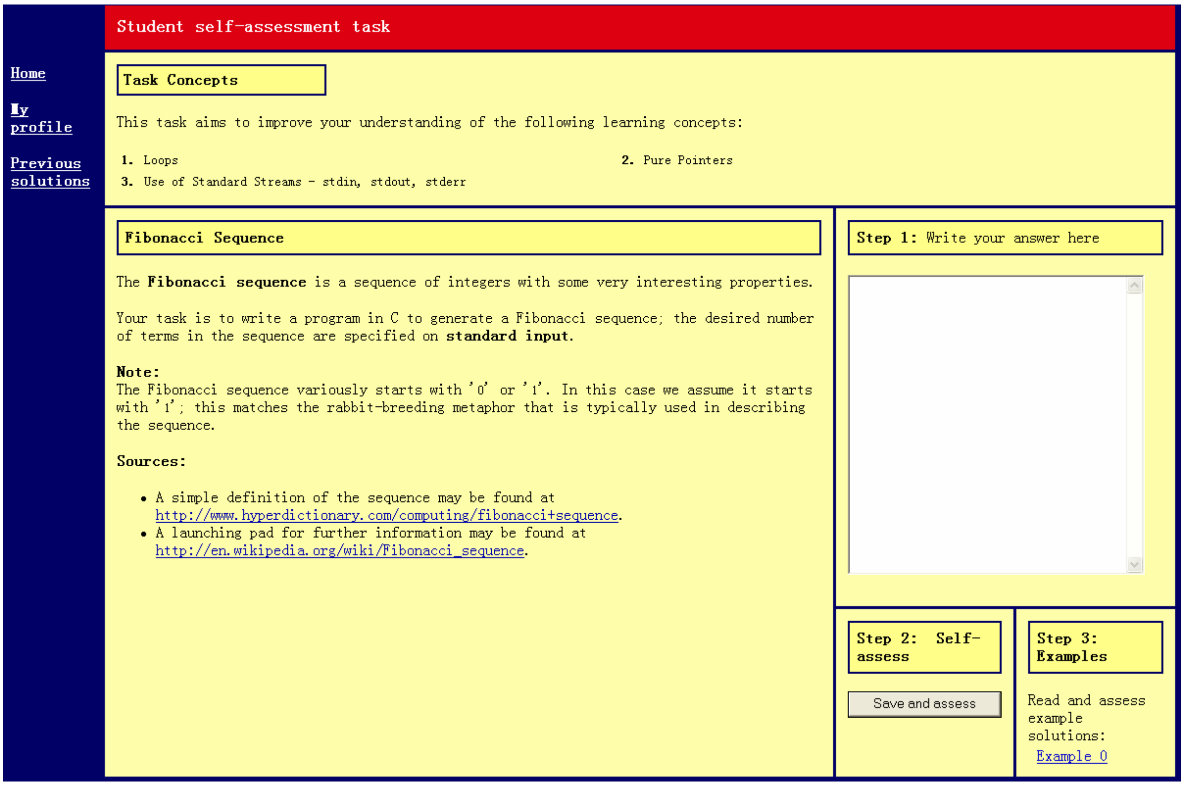


Figure 1 Problem Statement

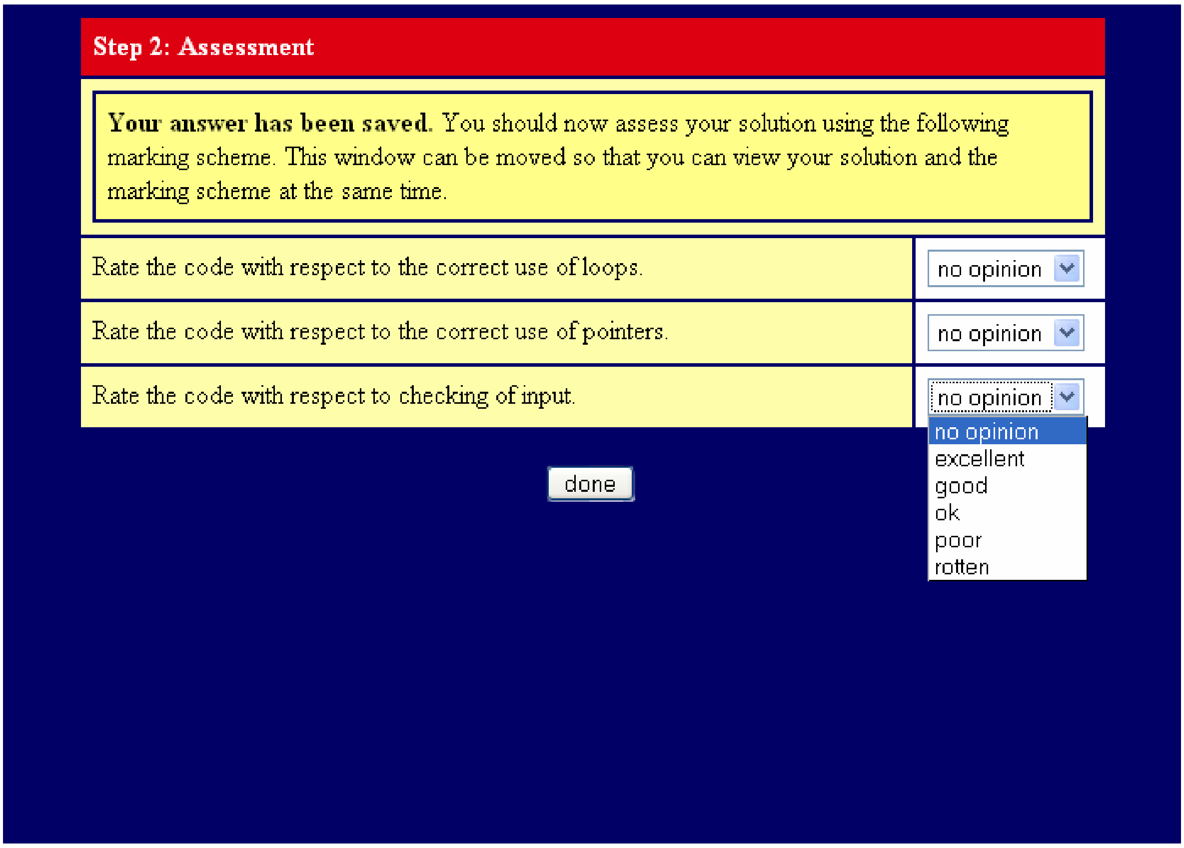
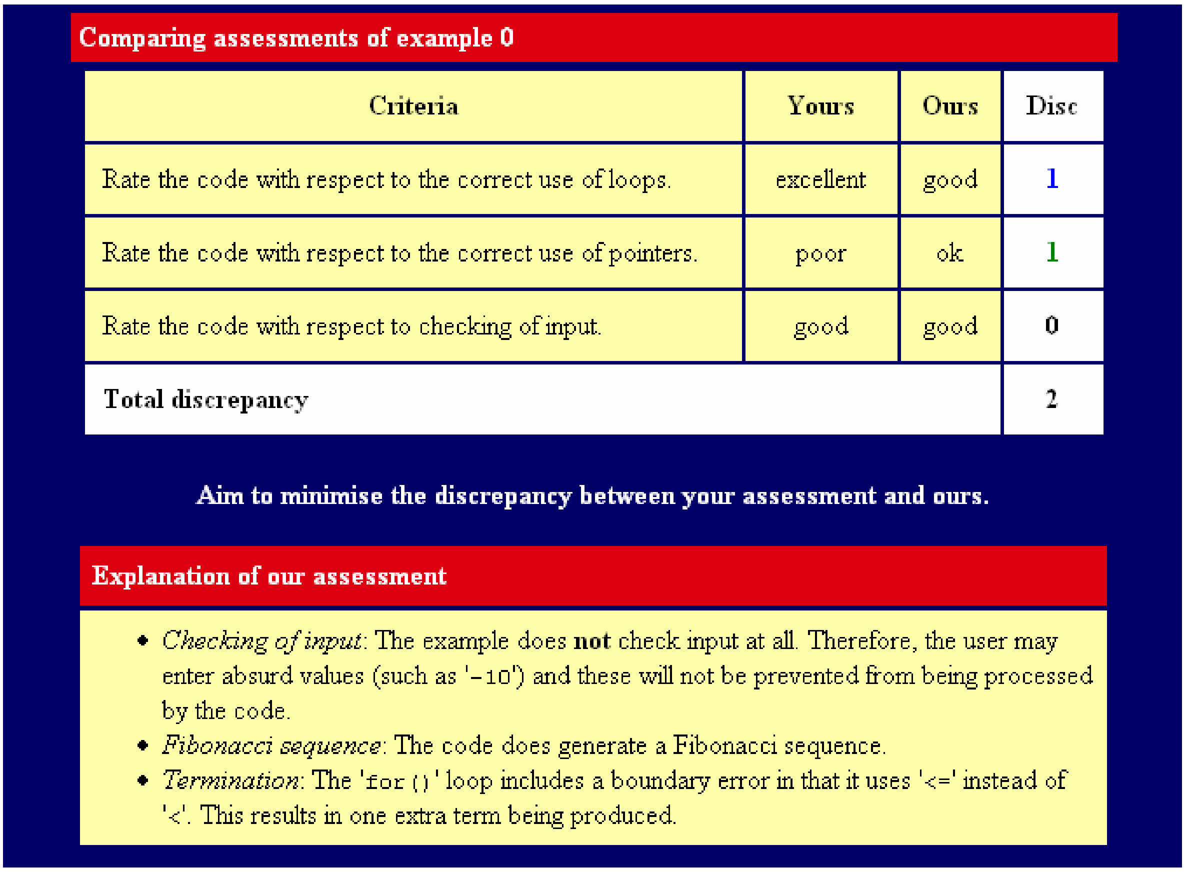


Figure 2 Solution Assessment

In this final stage, Assess also provides example solutions for students to assess. They are not necessarily ‘perfect’ solutions, but they do provide students some ideas to think about and evaluate. We usually create these examples, which demonstrate common misconceptions, poor style and similar elements, after grading final exam questions. We also strive to provide multiple examples, so we can illustrate different ways to do one task. These example solutions have been pre-assessed by teachers. The student assesses these solutions just as she did for her own solutions. Her assessment is then compared to the teacher’s assessment of the same example solution (Figure 3). The comparison gives the student feedback on:

- How the teacher marked the example;
- The difference between the teacher’s assessment and the student’s assessment;
- And why the teacher assessed it that way.

As the marking schemes used to assess the solution are consistent with the learning objectives of the task, the discrepancy between the student’s and the teacher’s assessments can indicate how well the student achieved the learning objectives. The probability that the student has understood each learning objective is then estimated and recorded in her student model, updating the system’s belief of how well she understands the learning objective. This information is used to illustrate the student’s learning progress to promote learner reflection-on-action, as upon seeing this information, she may explore why the teacher thinks differently from her and learn how to think as the teacher does, so to achieve a new level of understanding of the issues.



Comparing assessments of example 0

Criteria	Yours	Ours	Disc
Rate the code with respect to the correct use of loops.	excellent	good	1
Rate the code with respect to the correct use of pointers.	poor	ok	1
Rate the code with respect to checking of input.	good	good	0
Total discrepancy			**2**

Aim to minimise the discrepancy between your assessment and ours.

Explanation of our assessment

- *Checking of input*: The example does **not** check input at all. Therefore, the user may enter absurd values (such as '`-10`') and these will not be prevented from being processed by the code.
- *Fibonacci sequence*: The code does generate a Fibonacci sequence.
- *Termination*: The '`for()`' loop includes a boundary error in that it uses '<=' instead of '<'. This results in one extra term being produced.

Figure 3 Comparison and Discrepancy Values

Once a student finishes assessing example solutions, hence, completing the three steps of self-evaluation, she can view her user profile (See Figure 4). This displays the student's learning progress on a concept-by-concept basis in SIV [6] at the left. To the right, there is a list of hyperlinks for each learning goal. If the student clicks on one of them, statistics about the concept are presented in a new page, as shown in Figure 5. This information includes the student's mark for the concept, the number of times the student has attempted it, the class' average mark for it and the tasks that have it as a learning goal.

The user profile promotes reflection-on-action as it shows students their learning progress by making their student models inspectable with SIV [7]. It allows each user model to be explored in certain ways, such as selecting, deselecting and expanding a user model node and displaying the model accordingly. In the SIV display, teaching/learning goals are displayed with different indentations and in different colours. In particular, it shows what they know (i.e. the learning goals are displayed in green, e.g. the concept *Heap Memory* in SIV in Figure 4); how well they know it (i.e. the saturation of colours, e.g. the student knows *Linked List Creation* better than *Linked List Insertion*); what they do not yet know (i.e. the learning goals that are displayed in red or yellow, e.g. *Sorting and Searching* or *Preprocessor Techniques*); and how to learn it (i.e. what tasks aim to teach the concept, e.g. in Figure 5, the student can do *Task 1* to learn about the *Use of Constants for Boundaries*). From this information, students are encouraged to think about what they have learnt and how they have learnt it, i.e. reflecting on their experience. Furthermore, students are able to compare the system's beliefs about their C programming knowledge with their own beliefs. In this way, reflection is further encouraged, especially if the system's beliefs and the student's own beliefs differ. SIV can also show students what and how the learning concepts are related by displaying them in different sizes, with larger ones more closely related to the currently selected concept, which in Figure 4 is *Core Material*. It then becomes possible for students to improve their understanding of one concept by working with closely related concepts.

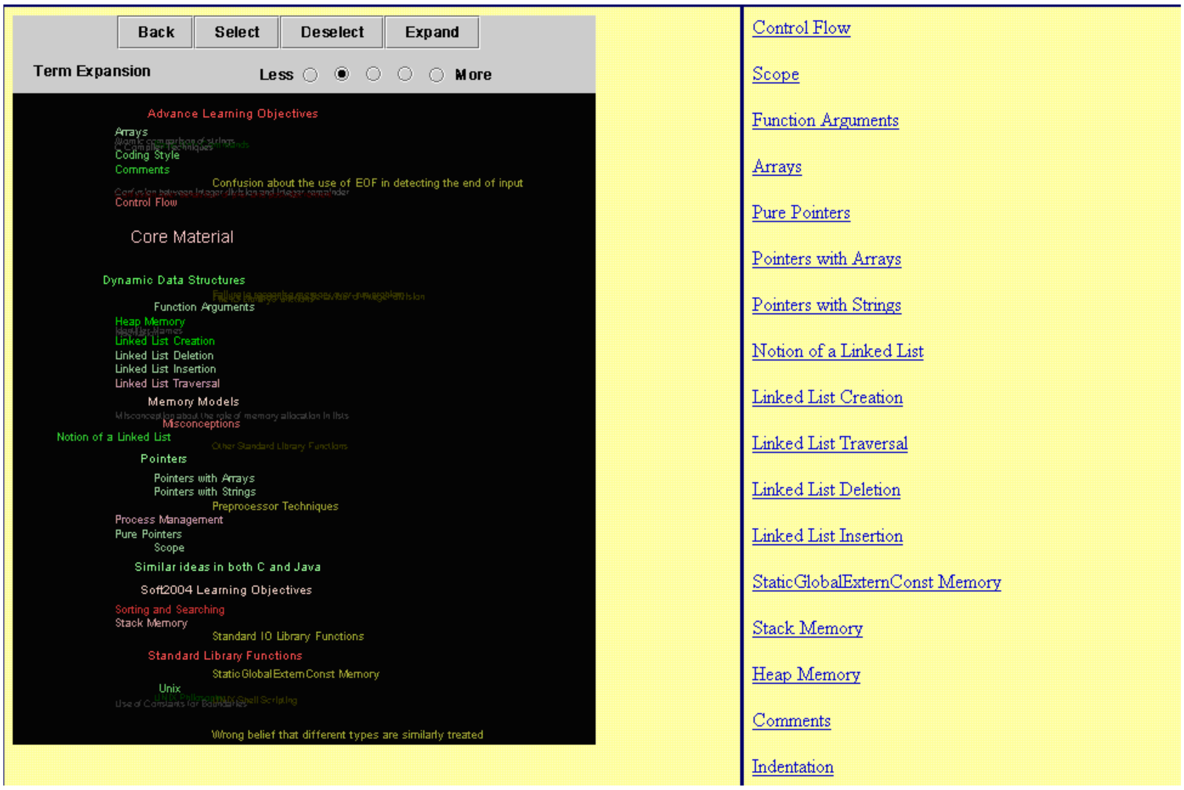


Figure 4 Student Feedback with SIV

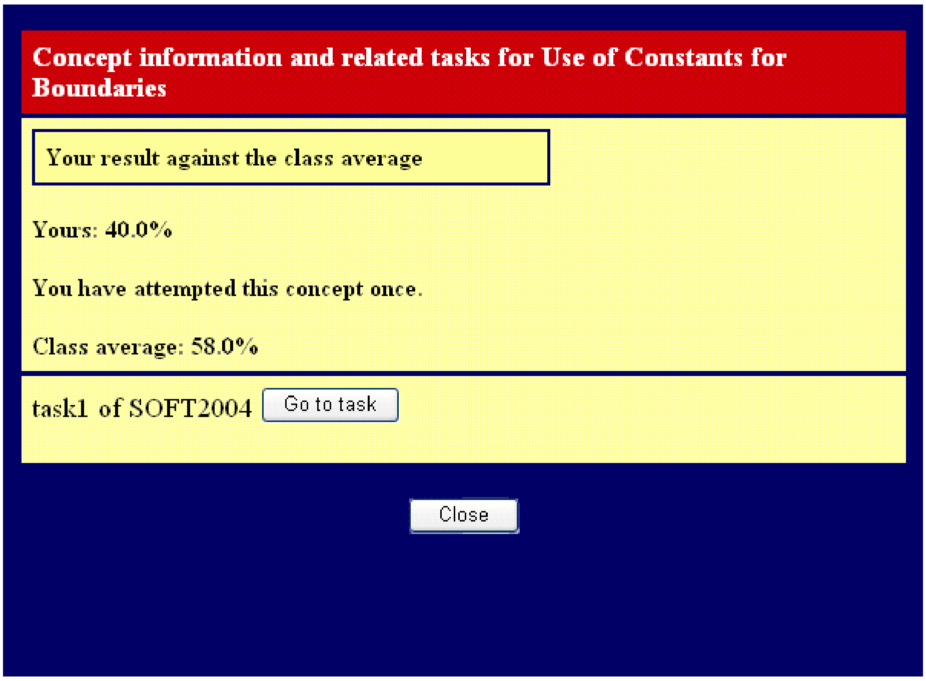


Figure 5 Useful Statistics

In summary, learner reflection-in-action and reflection-on-action, facilitated by student models are supported by the process of assessing solutions, reviewing overall progress with SIV and reviewing detailed performance comparison against the class.

3 Evaluation

We have conducted a preliminary user trial. Essentially, we wanted to find out if students could understand the learner model visualisation screen to appreciate the educational merits of it and use it to help them learn.

To gather information of students' opinions about reflection and elements of Assess, we asked them to rate the following questions on a scale of from 1 (meaning not important) to 6 (meaning very important):

1. *How important is it to reflect on what you know?*
2. *How important is it to study code that illustrates common mistakes students make?*
3. *How important is it to know the learning goals of a subject?*

We asked these questions to support our interpretation of the user's reaction to the interface.

The students' responses are presented in Table 1. Twenty-four students participated in the experiment. Most of them were undergraduates who had completed the Software Development Methods I course in 2004: four students obtained distinction plus (i.e. top 12% of the class, denoted with D+ in Table 1); nine students achieved credit (i.e. top 33%, represented by C); three students just passed (denoted by P); three students failed (denoted by F) and five students had not undertaken the subject (denoted by N), but four of them had done the preliminary programming subjects. There was only one student without any programming background. Most participants have experience with the Assess system without the knowledge layer, but none of them have used the new system. Results are grouped according to these grades in the table. Data from the table is illustrated in Figure 6 for better comparison.

Table 1 Students' Opinions on Several Aspects of Learning

Questions	N (n=5)	F (n=3)	P (n=3)	C (n=9)	D+ (n=4)	Overall	Standard Deviation
1. How important is it to reflect on what you know? (1 is not important at all, 6 is important)	3.8	4	4.3	4.1	5	4.2	0.46
2. How important is it to study code that illustrates common mistakes students make? (1 is not important at all, 6 is important)	4	3.3	5	4.8	5	4.4	0.75
3. How important is it to know the learning goals of a subject? (1 is not important at all, 6 is important)	3.2	5	4.7	4.6	5.8	4.7	0.94

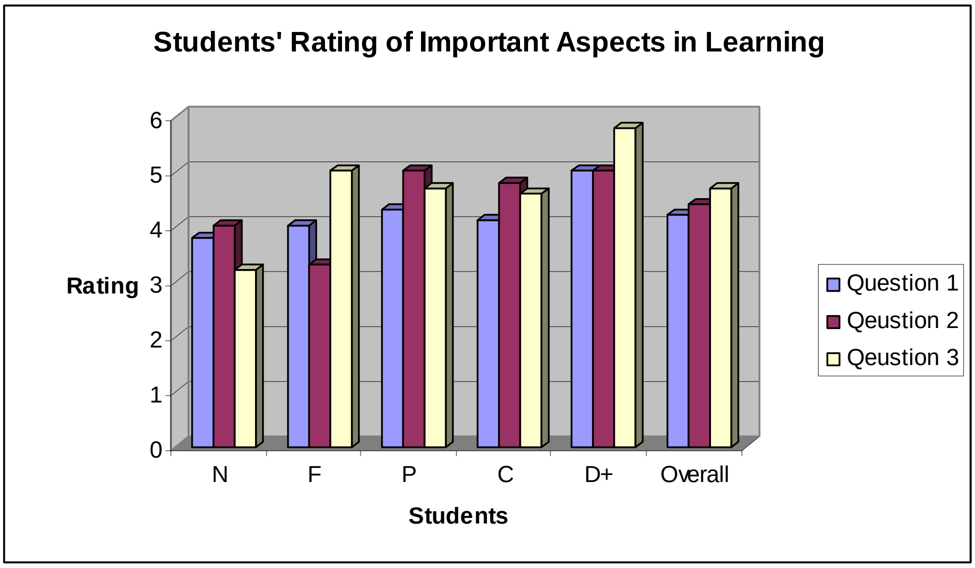


Figure 6 Students Rating of Important Aspects of Learning

The above figure shows the following:

- Students with higher grades, especially the D+ group regard the three learning techniques as more important;
- Forty-six percent of participants rated learner reflection as the least important of the three techniques;
- Students who failed the subject rated Question 2 as considerably lower importance compared to the other students. This means they do not believe understanding common misconceptions can help them learn.

Overall, the results indicate that the students think reflection is important and it is important to know the learning goals of a subject. Therefore, they should be able to appreciate that Assess can help them learn by encouraging reflection and providing a clear view of a subject's learning goals in the context of each learning task. Most students also think it can help them code better if they understand common coding misconceptions, which further justifies our decision to model students' misconceptions. Of course, another reason for including incorrect code is to make the self-evaluation process more interesting.

In the second part of the user trial, students were asked to examine an anonymous person's user profile in Assess and:

1. Find things that this person knows well;
2. Find things that the person is doing badly and;
3. Determine if the person is performing better than the rest of class on *Control Flow*.

The anonymous person's user profile is shown in Figure 7. To ensure the consistency of the results, we made all the tasks in the system inaccessible to students, so that they could not attempt any task during the user trial; this prevented them from modifying the system's beliefs of the anonymous person's knowledge state.

Note that we used a single anonymous person's Assess system usage for two reasons. Firstly, it meant that all participants saw exactly the same system usage and hence should have given comparable answers to the questions. Secondly, it was not feasible for us at the time to organise long term use of a suitable subset of Assess tasks so that substantial authentic learner models could be built.

All students were able to correctly answer Question 1 and 2, by identifying the learning concepts that the anonymous person understood and those concepts where the person lacked understanding. These students seemed to be able to use SIV effectively to judge what a student is modelled as knowing or not. This is consistent with evaluations of an earlier experiment, which evaluated this type of user model visualisation [6].

To answer the third question correctly, the students need to use the hyperlink of the concept, *Control Flow*, which is located to the right of SIV (See Figure 4), and view its statistics (See Figure 8), which shows the person is doing much better than the rest of the class. While twenty-one of the participants answered the question correct, three of them concluded that the anonymous person was performing worse than the class. This could be attributed to the possibility that the three students were trying to obtain this information directly from the SIV display but ignored the hyperlinks that were next to SIV. This finding suggests that we should explore improving the interface, by making a click on a SIV concept takes the student directly to the information about that concept, perhaps presenting information like that in Figure 5 in the area to the right of the SIV display.

Overall, 88% of participants were able to answer all the questions correctly. The students' answers indicate that they could understand the user profile interface. None of them had seen the interface before the user trial.

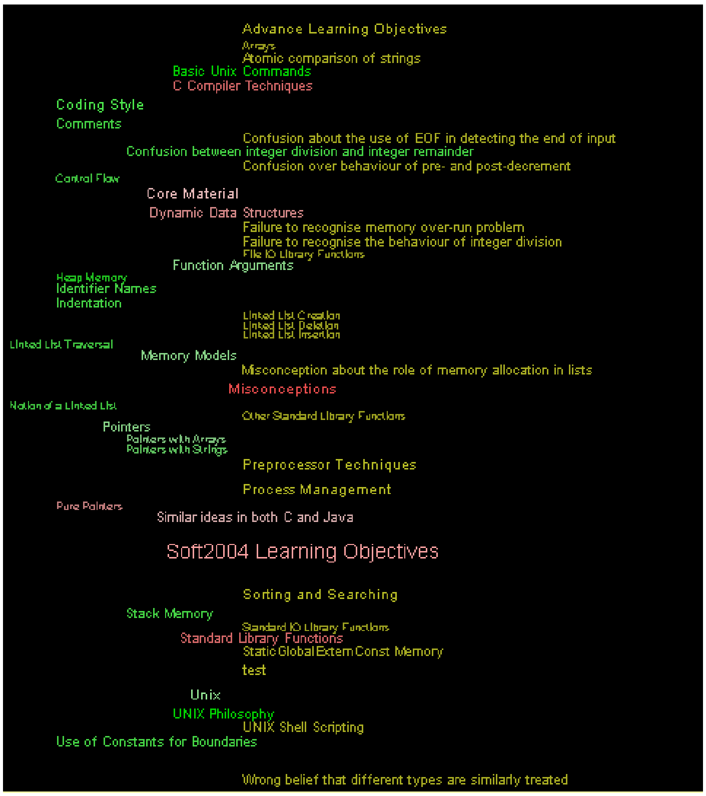


Figure 7 Anonymous Student's User Profile (Focus is on SOFT2004 Learning Objectives, more related concepts are larger, green means the student knows, red means the student does not know, indentation indicates how many time the student has attempted the concept. There are altogether 48 concepts in the system.)

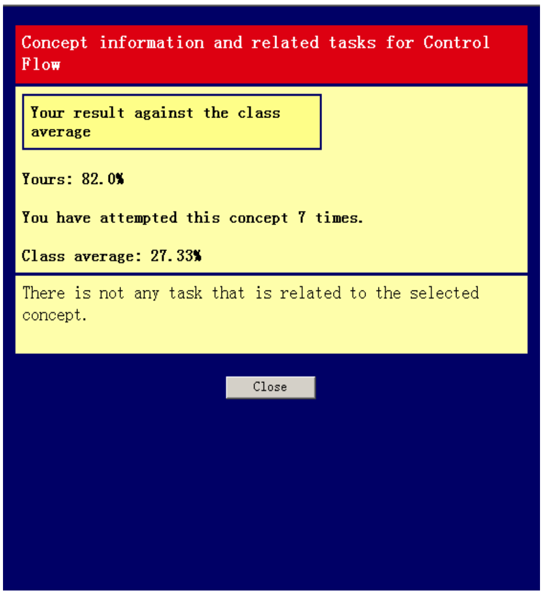


Figure 8 Anonymous Student's Statistical Information on Control Flow

4 Future Work and Conclusion

There are two directions we would like to explore in the near future. The first is to address the lack of automatic evaluation of a student’s solution. If students’ solutions can be automatically evaluated, we can provide more concrete feedback, which should make this part of the system more useful to students. It can also provide evidence of student's knowledge state.

A second goal is to provide guidance based on pedagogical module. For example, we propose to add a teaching agent based on the Model for Operational Skill Training (MOST) approach (Personal communication with Dr. Kalina Yacef from the University of Sydney). This would allow us to determine a teaching strategy and execute that strategy as a sequence of short-term teaching tactics. With the existing user models, it is possible to adapt this strategy to individual students based on their knowledge state and select the most suitable learning path for a student. Since learner reflection mostly occurs when the learner feels challenged [1], a pedagogical module can then further promote learner reflection.

We have presented Assess, a student self-evaluation system designed to support and promote learner reflection. Two types of learner reflection, namely reflection-in-action and reflection-on-action are supported by allowing the student to assess solutions and by providing informative learning progress feedback. We also described a preliminary user trial, which indicates that students were able to effectively use SIV and comprehend the information it provides and that they valued the availability of clear goals and appreciated the importance of reflection, two essential elements supported by Assess. The study also indicated strong appreciation of the value of studying misconceptions, one of the elements of the parts of Assess where students read and assess a set of supplied solutions to the tasks.

References

[1] Boud D., Keough, R., & Walker, D. (eds), 1985, Reflection: Turning Experience into Learning, Kogan Page, London.

[2] Boud, D., R. Keogh, and D. Walker. 1985. "Promoting Reflection in Learning: a Model," in *Reflection: Turning Experience into Learning*. Edited by D. Boud, R. Keogh, and D. Walker, pp. 18-40. London: Kogan Page.

[3] Schon, D., 1983, *The Reflective Practitioner*, Basic Books, USA.

[4] Fekete A, Kummerfeld B, Kay J and Greening T: *Identifying and improving the learning of borderline students in SOFT2004.* TR 545. March 2004, ISBN 1 86487 627 1.

[5] Kay, J., 1999, *A scrutable user modelling shell for user-adapted interaction,* PhD Thesis, Basser Department of Computer Science. Sydney, University of Sydney.

[6] Uther, J., 2001, *On the Visualisation of Large User Model in Web Based Systems,* PhD Thesis, University of Sydney.

[7] Zapata-Rivera, J.-D. and J. E. Greer, 2001, *Externalising Learner Modelling Representations,* AI-ED 2001 Workshop, External Representations in AIED: Multiple Forms and Multiple Roles, San Antonio, Texas.

Supporting Student Reflection in an Intelligent Tutoring System for Logic Programming

Josephine TCHETAGNI[1,2], Roger NKAMBOU[1], Jacqueline BOURDEAU[2]
[1] *GDAC, Computer Science Department, UQÀM, Montréal, (QC) H3C 3P8 Canada*
[2]*LICEF Research Center, Télé-Université, Montréal, (QC) H2T 3E4 Canada*
{tchetagn@info.uqam.ca,nkambou.roger@uqam.ca, bourdeau@licef.teluq.uquebec.ca}

Abstract. This paper presents a model of tutoring dialog which guides the learners through a focused and explicit reflection, in an intelligent tutoring system (ITS). The structure of this model is an interpretation of the fundamental components of reflective thinking as defined by Dewey. The contents of these dialogs refer to the nature of the skill on which the learner reflects, as defined in a taxonomy of skills. An integration of these dialogs in Prolog-Tutor, an ITS for Logic programming, is described and their original features are highlighted. Two issues related to this explicit approach to reflection are outlined: (1) the validity of this interpretation of Dewey's components of reflective thinking; (2) the characteristics of the learning outcomes from explicit reflection.

Introduction

A recent focus in student modeling in ITSs stresses the importance of involving learners in the diagnosis process upon which the tutoring interactions rely ([11]). This view is mainly supported by a set of approaches which foster learners' reflection on their own learning processes. Schön[21] describes two types of reflection: reflection-in-action and reflection-on-action. Reflection-in-action occurs during the completion of a task: it allows someone to reshape what is worked on, while working on it. Reflection-on-action occurs when examining one's own learning process retrospectively.

Learner' models are used as tools to elicit reflection-on-action. For example, learners can examine their level of mastery of the skills or of their coverage of the field ([3, 26]). Reflection can also consist in viewing the model of a peer learner, in order to support collaborative learning ([25]). Learners can also reflect by viewing and editing their model in order to change or to negotiate its content ([3, 4]).

Tutoring dialogs are generally the preferred tactics to support reflection-in-action in ITSs: with natural language processing components (Auto-Tutor[17], Atlas-Andes[20], Circsim[8], CycleTalk[19], Geometry-Explanation-Tutor[1], Why-Andes[14]) or with menu-driven driven components (CATO[2], Ms. Lindquist[10], STyLe-OLM[6]).

Two issues pertain to the use of tutoring dialogs in a learning situation. First, they should be conducted in a coherent manner so that the learner is always aware of the focus towards a learning goal or at least, towards a solution ([6]). Second, the reflective activities which emerge from these tutoring dialogs are a side effect of their interactive nature. There is not always a concrete evidence of their occurrence ([6]). Indeed these dialogs should not only allow the learner to construct the right solution to a problem, they should explicitly elicit the skill which underlies the construction of that solution ([1]).

None of the previous systems explicitly consider these two issues at once. Their tutoring dialogs allow the learner to build the desired answer. However the question remains: what happened exactly? Was it assimilation of knowledge, guessing of the right answer or an explicit insight on the skills to be learned and used? Auto-Tutor's tutoring dialogs are based on a formal structure which fails to warrant reflection about a skill. Atlas-Andes uses knowledge construction dialogs (KCD) to guide the learner towards a solution. However the nature of what is really processed can not be conclusively stated from these dialogs: the KCD does not articulate the knowledge element or the skill to which it corresponds (*traced* from a task graph in Andes). Geometry-Explanation-Tutor addresses these issues only in the context of reflection-on-action: the learners self-explain their solution to a geometry problem by explaining in their own words how they have applied a principle.

Tutoring dialogs could embed goals which *explicitly* aim to foster students' reflection. Two main properties are desirable in such dialogs: (1) their structure should warrant that they coherently and continuously focused towards a learning goal; (2) their contents (the communicative acts) should enable *concrete evidence* that, besides leading the learner to the successful construction of the right answer to a problem, they allow him to acquire the skills which were targeted. One approach to achieve this is by interpreting a theory of reflective thinking, while modeling these tutoring dialogs. Historically, the challenge of defining reflection has been studied by several scholars. Lewin's model of action research is oriented towards reflection during problem-solving, in social and organizational settings ([13]). It would be inappropriate to use this model in the context of one-on-one tutoring dialogs in ITS, since it intrinsically relies on the dynamics of social interactions. Another theoretical basis for reflection as a practice in education is the work of John Dewey ([5]). Dewey's theory of reflection is foundational since it relies on a logical and philosophical argument, to explicitly articulate the components of reflective thinking. Moreover, it has been acknowledged, since it inspired the main contemporary models of reflection, whether related to experiential learning (Kolb[12]) or to professional education (Schön[21]).

The purpose of this paper is to model tutoring dialogs, which interpret the fundamental components of reflective activity, as stated by Dewey, in order to explicitly support learners' reflection. These dialogs are instantiated in Prolog-Tutor, an ITS for Logic programming with the Prolog language. Section 1 describes the components of Dewey's reflective thinking. Section 2 describes Prolog-Tutor. It also describes how Dewey's reflective thinking components have been interpreted in order to design a tutoring dialog which explicitly fosters reflection. Section 3 illustrates the implementation of these dialogs in Prolog-Tutor. Future work is briefly outlined in Section 4.

1. An Explicit Consideration of Reflective Thinking

According to Dewey, reflective thinking is neither the occurrence of ideas, nor the occurrence of judgment. Reflection lies in the way in which the belief about a conclusion has been constructed. That conclusion has to be drawn *on the grounds of reasoning* and on a set of *evidences* which support that reasoning. In this vein, reflective thinking integrates five main components: (**1)** an experience with a contradictory/ controversial/obscure situation where the subject must reason in order to draw out an acceptable conclusion/explanation**; (2)** the process of intellectualization triggered by the controversial situation which consists in the formal definition/identification of the conditions (the facts) which cause the contradiction/controversy/obscurity; **(3)** the process of proper reflection which consists in pondering on the facts of a situation, in order to reach an acceptable conclusion; **(4)** The reflective learner can also enter into a process of internal reflection which consists in relating ideas in order to infer further hypotheses; **(5)** the result of reasoning after considering a hypothesis, which consists in selecting a single conclusion amongst those that emerged. That conclusion could be eventually tested if it does not consistently follow from the facts which characterize the situation.

The goal of this paper is to interpret these prescriptions in a tutoring dialog in Prolog-Tutor.

2. Logic Programming and Prolog-Tutor Dialog Models

In Prolog-Tutor, the learning experience is designed to focus on the acquisition of basic skills related to the *paradigm* of Logic programming, namely: (1) understanding the intuition, rationale and use of a `knowledge base`[1]; (2) understanding the intuition of `unification` and applying the corresponding principles in a concrete situation; (3) understanding the intuition of the `resolution (or proof) of a goal` and applying the corresponding principles in order to `perform` a concrete `resolution`. The implementation of Prolog-Tutor distinguishes knowledge elements which refer to the contents of the domain to be learned, from the skills to be acquired in that domain. A *skill* comprises an *ability* that must be *applied to a knowledge element* ([15]). This approach has been adopted in order to explicitly formulate the different points of view from which a knowledge element can be viewed. For example, to acquire the `resolution of a goal` as a concept (or to "`understand the resolution of a goal`"), it is sufficient to "`understand the unification`". To acquire the `resolution of a goal` as an operation or a procedure (or to "`apply/perform the resolution of a goal`"), one has to "`understand the resolution`" and to "`perform/apply the unification`". The next section describes the design of the tutoring dialogs which supports learning through reflection in Prolog-Tutor.

2.1. Learning to "`Perform the Resolution of a Goal`" in Prolog-Tutor

In order to acquire the skill pertaining to "`perform the resolution of a goal`", the learner must solve a problem where the task is to determine the results of the `resolution of a goal`, given a Prolog `knowledge-base` (Table 1).

If the learner's answer is incorrect, the tutoring strategy consists in breaking-down the problem into sub-problems. These sub-problems are presented through a tutorial dialog with the learner. Each sub-problem corresponds to a particular stage in the process of "`performing the resolution of a goal`". At each stage, the learner is asked a question which has two pedagogical goals. First, each question should be suggestive enough so that it fosters a hidden (implicit) reflection which in turn could remedy the cause of the learner's error: re-examining the data pertaining to the current state of the problem, recalling a principle pertaining to the domain, applying a principle in order to determine an answer. Second, this approach allows the tutor to achieve cognitive diagnosis in a more tractable way, since each question (a diagnostic question or DQ) is explicitly associated with the elicitation of a particular skill. Table 1 shows that the learner is asked a question which requires the computation of a result. First, the question is suggestive as it indicates to the learner the first `Prolog-Rule` which will be used to `perform the resolution of the Goal` at hand: `grand_father(X,P):-father(X,P),father(Z,P)`. Second, in order to understand the question, the learner must reflect on the principle which is required to elicit the skill `use(Prolog-Rule)` in `the resolution of a GOAL`:

[1] For the purpose of clarity, the font ``Courier New`` is used to refer to the expressions related to the Prolog language or to Prolog-Tutor.

```
[IF the HEAD of a PROLOG-RULE is UNIFIABLE with the GOAL to resolve, THEN use
that PROLOG-RULE to perform the RESOLUTION]
```

Finally, in order to answer the question, the learner must reflect on the principles which underlie the skill: "manipulate(Prolog-Rule) in the resolution of a goal".

Table 1: Tutoring dialog in Prolog-Tutor with implicit reflection

<table>
<tr><td colspan="2">Learning situation:
Give the solution to the resolution of the GOAL <code>grand_father(X,joseph).</code> Formulate your solution as if you were the Prolog-Interpreter, using the given knowledge base</td></tr>
<tr><td colspan="2">The learner gives an erroneous answer. Tutor starts a dialog to construct the solution with the learner.
<u><code>Knowledge Base</code></u>
<code>grand_father(X,y):-father(X,Z), father(Z,y)</code>
<code>grand_father(X,john):- grand_son(john,X).</code>
<code>grand_father(X,P):-father(X,P),father(Z,P)</code>
<code>father(steph, anton). father(anton,ken). father(anton, joseph)</code></td></tr>
<tr><td>Expected/Hidden Reflection on the P1</td><td>Illustration</td></tr>
<tr><td>[The first Prolog-Rule used is grand_father(X,P):-father(X,P),father(Z,P)] [Observation]

[grand_father(X,P):-father(X,P),father(Z,P)is used because its HEAD is UNIFIABLE with the GOAL] [Inference]</td><td>1. Tutor[Challenges, Asks]: What is the result of the first trial to resolve grand_father(X,joseph) with grand_father(X,P):-father(X,Z),father(Z,P)
2. Learner[Answers] Success of the resolution
2.1. Tutor [Positive feedback]: No. What is the result of resolving the first sub-goal father(X,Z)?</td></tr>
</table>

It is likely that a reflective process implicitly takes place through the use of this tutorial dialog. However, from a pedagogical perspective, this approach does not insure that the learner really elicits the skills that are targeted. This paper proposes that the components of reflective thinking may be used for this purpose.

2.2. Towards Dialogs Models for Explicit Reflective Thinking in Prolog-Tutor

When Prolog-Tutor diagnoses that a learner lacks knowledge about a particular skill, the goal is to elicit an explicit reflection pertaining to the acquisition of that skill. In Table 1 for example, the learner's answer at Step 2 is incorrect. The tutor starts a sub-dialog to lead the learner to construct the correct answer to the question asked in Step 1. In order to help the learner to reflect on a particular skill, this dialog is designed in such a way that it interprets Dewey's components of reflective thinking. This dialog model should also achieve a certain level of generality by interpreting Dewey's components on the basis of the nature of a skill rather than on its instance in the Prolog domain. A skill can be general enough (regardless of the context in which it is used) to be defined with respect to a skill taxonomy, which gives a principled description of its properties.

Table 2. Skills and objectives of reflective thinking (inspired from Drake[7])

Skill	Objective of reflective thinking
- Identify a concept	Determine the correctness of a classification ; verify the meaning; recognize the attribute; justify the wrongness of a classification
-Apply,Use principles	Verify statement confirms/contradicts; Identify premises /consequences.
-Verify Arguments	End up with contradiction/confirmation; Determine assumptions/the consequences.

The following method has been selected for this interpretation: (1) determine the goal of the reflective process with respect to each type of skill; (2) infer how (interpret) the components of reflective thinking can be used to reach these goals. Concerning the goal of reflection on a particular type of skill, Dewey outlined that the intellectual activities which underlie the process of judgment are analog to reflective thinking. Therefore, Drake's ([7]) description of the goals of judgment is relevant, since these goals are defined specifically for concepts, principles and arguments (Table 2).

Table 3. A model of explicit reflection on a *principle*

Reflect on the Application of Principle P1 Goal of the reflection: Determine whether a statement confirms or contradicts P1	
Stages	**Interpretation in the Context of 'Applying a Principle'**
Situation	1- Describe several facts to which the principle could be applied. 2- Outline the possible facts which could be drawn from of the situation
Reasoning about the principle	1- Make a statement S which corresponds to the targeted conclusion. 2- Verify if P1 contradicts or confirms S.
Evaluation (Optional)	1- If S is an acceptable conclusion, no need for evaluation. 2- If S is an unacceptable conclusion, show the contradiction to the learner by having him evaluating S with P1.

Next, a dialog model for each targeted skill has to be designed in order to reach the goal of reflection. The interpretation of the reflective process uses the characteristics of a skill as defined in a taxonomy. Using Gagne's taxonomy ([9]), a model of explicit reflective thinking about a principle is presented (Table 3). This model stems from the fact that a principle is invoked in two ways: (1) when the conditions to which it applies are present; (2) when the consequences that it implies are present.

The following section describes the implementation in Prolog-Tutor of the tutoring dialog which fosters implicit reflection. The integration of a tutoring dialog for explicit reflection about a skill is also discussed.

3. Towards Explicit reflection in Prolog-Tutor: ER-Prolog-Tutor

Tutorial dialogs are already implemented in Prolog-Tutor. This section describes how explicit reflection can be integrated into these dialogs.

3.1. The basic tutoring dialog in Prolog-Tutor

In Prolog-Tutor, the basic tutoring dialog (noted **D1**) is triggered when the learner gives an incorrect solution to a problem. It consists in asking questions leading to the construction of the correct solution. Each question is explicitly associated to a skill to indicate that the elicitation of that skill is required for answering that question correctly. Therefore, these questions are considered as diagnostic questions (DQ). Each DQ in D1 is formulated in such a way that it is sufficiently suggestive to support an implicit reflection on the corresponding skill, which in turn could remedy the cause of the learner's error. Moreover, the learner's answer to a DQ determines how the dialog evolves: if it is correct, the next DQ is addressed; if it is incorrect, the tutor focuses the interaction on the skill associated to that DQ. This approach implies that each interaction in D1 can be seen as a loop between remedial instruction and cognitive diagnosis ([23]). The diagnosis process is supported by asking suggestive remedial DQs. The result of this diagnosis determines in turn the next DQ that is addressed, and so on.

The tutor controls the interaction with the learner, in order to *maintain the focus* towards the pedagogical goal of acquiring a skill, while constructing the solution to the problem. The tutor may: (1) ask the learner a question in order to diagnose the skills that are lacking; (2) provide positive or negative feedback according to the learner's answer; (3) give instructions to the learner about how to answer a DQ; (4) provide an explanation regarding the right answer to a DQ; (5) provide an explanation about the rationale of a DQ (what is the purpose of asking that DQ in the current problem solving context?); (6) summarize the sub-dialog which corresponds to a DQ (addressed in the next section), in order to outline what was analyzed in it.

Besides answering DQs, the learner can ask three types of questions. He can express the fact that: (1) he does not understand the DQ; (2) he does not know how to use the interface in order to answer a DQ; (3) he does not understand the rationale of the DQ in the context of the current problem. From a pedagogical perspective, the third type of question is the most important while the others could be considered as requests for hints. Indeed, the main pedagogical goal of using tutoring dialogs is to understand the learner and to have him acquire or deepen the acquisition of a skill. In this context, it is important that he always understand the line of reasoning which characterizes the tutoring dialog and its relation with the exercise currently at hand. The third type of questions responds to this need.

If the learner fails to answer a DQ correctly, it shows that he lacks the skill associated to that DQ or that he has not understood the DQ. However as stated above, the learner has the opportunity to express the fact that he does not understand a question. The tutoring strategy here consists in focusing the learning process on that skill, rather than continue constructing the solution to the problem. When a skill comprises sub-skills, Prolog-Tutor can use two tactics in order to determine the sub-skills that need to be focused on. The sequential tactic consists in focusing on all the sub-skills one after another. The "most-probable-explanation" tactic consists in addressing the sub-skills so that their lack best explains the lack of the diagnosed skill ([16]). When a skill does not comprise sub-skills, Prolog-Tutor directly focuses on it.

Prolog-Tutor focuses reflection on a skill by asking the learner questions, which are specifically related to that skill. This may happen outside the context of the problem currently being solved, by generating an exercise which requires specifically the elicitation of the diagnosed skill. This may also happen in a sub-dialog in the context of the problem currently being solved. Our goal is to design this sub-dialog so that it triggers explicit reflection. The next section introduces Explicit-Reflection-Prolog-Tutor (ER-Prolog-Tutor), an extension of Prolog-Tutor where tutoring dialogs support explicit reflection.

3.2. Integrating Explicit Reflection into Prolog-Tutor

When the learner fails to answer a DQ, the tutor generates a sub-dialog which focuses on the corresponding skill. This sub-dialog (noted **D1.2**) is intended to engage the learner in an explicit reflection on the skill which corresponds to that DQ.

In Figure 1, the tutor asks the learner two questions. These questions are related to (the skill) `performing a resolution of a GOAL or of a FACT in Prolog: [what is the goal or the fact to prove in this resolution?] [what is the first element of the prolog knowledge base which corresponds to that goal?].` If the learner fails to answer the second question, the system: (1) diagnoses that the skill required to answer that DQ may be lacking: `[manipulate(PrologRule) in the resolution of a goal];` (2) generates a sub-dialog in order to provide the learner with an insight into this skill through explicit reflective thinking.

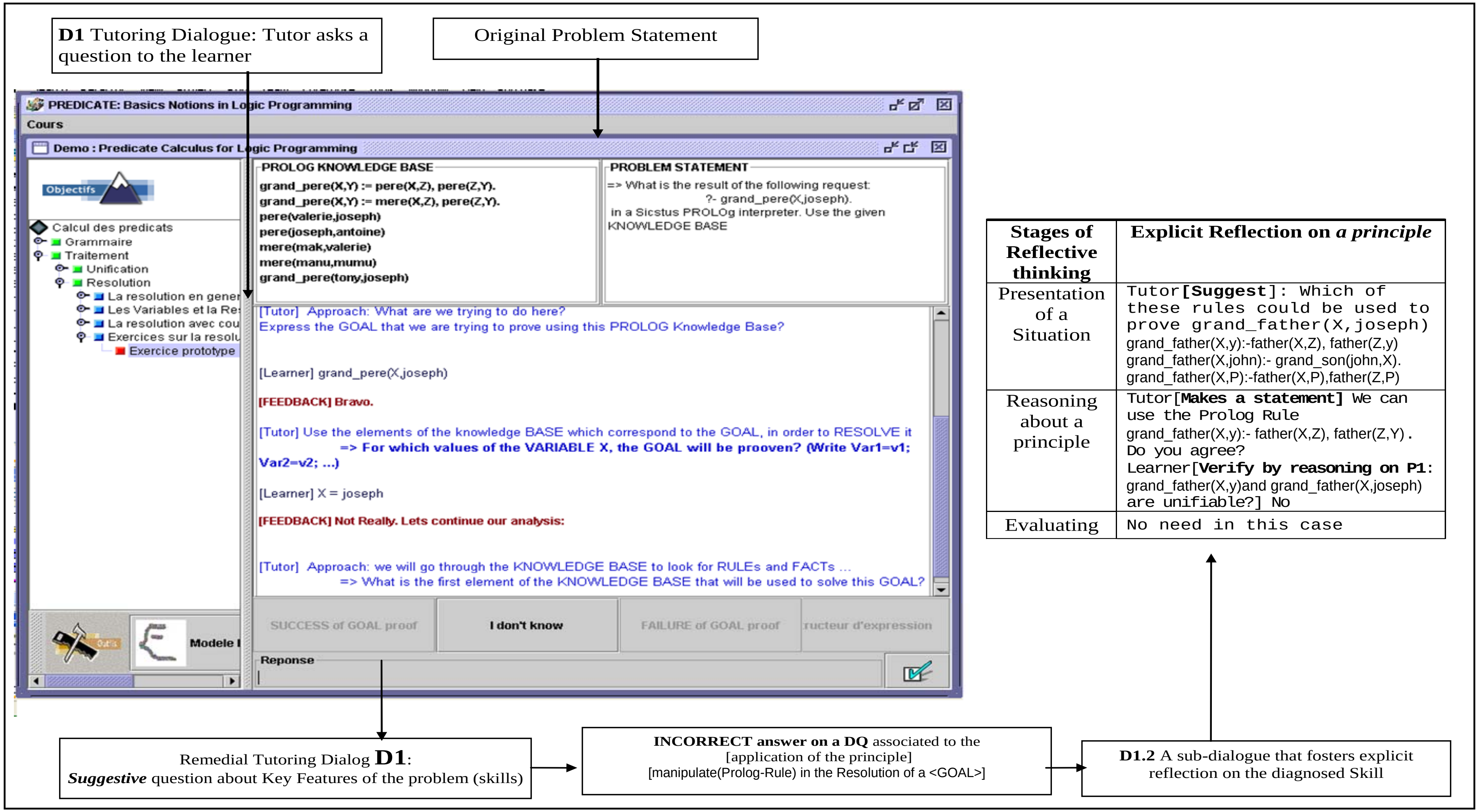


Figure 1. Shifting the tutoring dialog from D1 to D1.2

The sub-dialog illustrated in Figure 1. tackles reflection on the first sub-skill of [manipulate(PrologRule) in the resolution of a goal], which is [use(PrologRule) in the resolution of a goal]. The tactic consists in generating a specific situation where the learner has to use a prolog-rule in the context of performing the Resolution of a Goal. In this case, the tutor asks the learner to choose the prolog-rule which should be used to solve the current Goal. The situation is "controversial" in the sense that the alternatives are similar. Providing a good answer necessitates a reflection on the principle, which underlies it. When the learner provides a wrong answer, the tutor tries to have him elicit the principle which underlies the "use of a prolog-rule" when "performing the Resolution of a Goal".

A learner could also reflect on a skill inside the context of the exercise in which it has been diagnosed, as when the tutor asks: "what is the rule of the knowledge base that will be used for this resolution?", "how will you use it in association with the Goal to prove?"

The dialogs have two pedagogical goals. First, each DQ in D1 allows the system to directly diagnose learners' difficulties. Second, each DQ in D1.2 allows the learner to reflect (explicitly) about the skills that are identified as lacking. This dialogue nesting is relevant because: (1) the system performs an interactive diagnosis which fosters an implicit reflection; (2) when the system diagnoses the lack of a particular skill, it triggers a process which allows the learner to reflect upon the skill more explicitly, using the model described in Section 2; (3) each question (either in D1 or in D1.2) is associated to a skill so that the cognitive diagnosis is continuously refined

3.3 Reflection in Action in Prolog-Tutor: Contribution

Prolog-Tutor supports implicit reflection-in-action by triggering remedial tutoring dialogs (D1 in section 3.1) when the learner does not give the right solution to an exercise. In this dialog, questions are asked to the learner in order to guide him towards the right solution. In that perspective, Prolog-Tutor shares the features of ITSs where learning is supported using tutoring dialogs. The added value in D1 is that Prolog-Tutor can explain to the learner the goal/rationale of a DQ in the context of the problem that he is currently solving. This allows the learner to understand the line of reasoning which characterizes the tutoring dialog and its relation with the exercise currently at hand.

Most importantly, ER-Prolog-Tutor fosters explicit reflection-in-action by triggering remedial tutoring sub-dialogs (D1.2 in section 3.2), when the learner is unable to give the right answer to a DQ in the upper level dialog (D1 in section 3.1). The originality of ER-Prolog-Tutor is that the structure of the remedial sub-dialogs is based on the components of the process of reflective thinking as stated by Dewey. This structure insures that the learner is coherently guided towards the acquisition/understanding of a skill, and not only towards constructing the right solution of the exercises. Coherence is achieved since the structure of the dialog relies on Dewey's acknowledged theory of reflective thinking. The originality in ER-Prolog-Tutor also bears on the contents of the questions that are asked in the explicitly reflective sub-dialogs. These questions refer to the nature of the skill on which the learner reflects, as defined by a taxonomy, contextualized in the problem currently being solved ([24]). The benefit of this approach is that these dialogs models and these questions patterns are transferable to other learning domains, as long as these domains are clearly defined in terms of skills, derived from the same taxonomy.

Prolog-Tutor and ER-Prolog-Tutor foster reflection-in-action while continuously refining the diagnosis hypotheses about the skills that the learner lacks. Each question in the tutoring dialogs (D1 and D1.2) is explicitly associated to the skill required to answer correctly. The benefit of this approach is that as a tutoring dialog evolves, the more accurate the questions asked to the learner become. This approach also allows the learner to appreciate concretely the relationships among the skills of the domain, while trying to construct the right answer to the problem.

Learning the Prolog language has been addressed in ITSs ([18, 22]). Prolog-Tutor is distinct from these systems in that it teaches the *paradigm of Logic programming* with the Prolog language, and not the writing of algorithms in Prolog language.

3. Future Work

The next step of this research consists in analyzing two issues: (1) the validity of the interpretation of Dewey's components of reflective thinking; (2) the characteristics of the learning outcomes from explicit reflection.

The validity of the interpretation will be assessed by two experts: an instructional designer and an experienced lecturer in the Logic programming paradigm. First, they will fill out a questionnaire to assess the validity of the interpretation of the stages of reflective thinking. Second, given a learning trace extracted from ER-Prolog-Tutor, they will fill out a questionnaire to assess the validity of the implementation of these dialog models.

The characteristics of the learning outcomes from explicit reflection will be analyzed in a qualitative study. The goal is to match each reaction of a learner with a component of Dewey's reflective thinking, when being asked a question. Think-aloud protocols will be used in an experiment where the participants will have to interact with ER-Prolog-Tutor. For example when requested to formulate the solution to a Logic programming exercise, learners in D1-only could blindly try to apply the Logic-programming algorithms (`unification` and/or `resolution`) which support the generation of the solution; learners in D1+D1.2 could try to observe relevant data in the problem and then reason about it on the basis of the principles which dictate the generation of the solution. This qualitative experiment with Prolog-Tutor and ER-Prolog-Tutor is planned with ten undergraduate students in Computer Science. This experiment will show how the learners formulate their solution to Logic programming exercises after having experienced a learning session with the tutor.

Conclusion

This paper presented a tutoring dialog model to foster explicit reflection on a skill in ER-Prolog-Tutor, an extension of Prolog-Tutor. Prolog-Tutor already offers tutoring dialogs which foster implicit reflection. ER-Prolog-Tutor integrates tutoring dialogs which explicitly engage the learner in a reflective process *in action* on a targeted skill. The structure of these dialogs is based on an interpretation of Dewey's acknowledged theory of reflective thinking. The contents of these dialogs is based on an articulation of the skill on which the learner reflects; this articulation refers to the nature of that skill, as defined in a taxonomy, in the context of Logic programming. These two properties of the tutoring dialogs in ER-Prolog-Tutor constitute the main contributions of this paper. Future work consists in a qualitative analysis of ER-Prolog-Tutor. Two issues have been highlighted: (1) the validity of the interpretation of Dewey's theory according to the nature of the skill on which the learner will reflect; (2) the characteristics of the learning outcomes from explicit reflection on a skill. These issues will be addressed in qualitative experiments involving the following participants: an instructional designer, an experienced lecturer in the domain of the Logic programming paradigm and ten undergraduate students in Computer Science.

References

[1] Aleven, V., Popescu, O., and Koedinger, K. Towards tutorial dialog to support self-explanation: adding natural language understanding to a cognitive tutor. In Hoppe, U., et al. (*eds*), *Proceedings of AIED 2003*. IOS Press. p. 39-46

[2] Ashley, K., Desai, R., and Levine, R. Teaching Case-Based Argumentation Concepts using Dialectic Arguments vs. Didactic Explanations. In Cerri, S., et al. (*eds*), *Proceedings of ITS2002*. Springer-Verlag. p. 585-595

[3] Bull, S., McEvoy, T., and Reid, E. Learner Models to Promote Reflection in Combined Desktop PC / Mobile Intelligent Learning Environments. In Hoppe, U., et al. (*eds*), *Proceedings of AIED2003*. IOS Press. p. 199-208

[4] Bull, S. and Pain, H. Did I say what I think I said, and do you agree with me?: inspecting and questioning the Student Model. In Greer, J. (*eds*), *Proceedings AIED95*. AACE. p. 501-508

[5] Dewey, J., *How we think: A restatement of the relation of reflective thinking to the educative process*. Heath & Company,1933.

[6] Dimitrova, V. Diagnostic Interactions that Promote Learner's Reflection. In Hoppe, U., et al. (*eds*), *Proceedings of AIED 2003*. IOS Press. p. 228-237

[7] Drake, A., J., *Teaching Critical Thinking: Analysing, learning and teaching critical skills*. Interstate,1976.

[8] Evens, M., et al. CIRCSIM-Tutor: An intelligent tutoring system using natural language dialogue(*eds*), *Proceedings of MAICS 2001*. p. 16-23

[9] Gagne, R., *The Conditions of Learning and Theory of Instruction*. Holt, Rinehart & Winston.,1992.

[10] Heffernan, N. and Koedinger, K. An Intelligent Tutoring System Incorporating a Model of an Experienced Human Tutor. In Cerri, S., et al. (*eds*), *Proceedings of ITS2002*. Springer. p. 596-608

[11] Kay, J. Learner Know Thyself: Student Models to give Learner Control and Responsibility. In Halim, Z., et al. (*eds*), *Proceedings of ICCE97*. p. 17-24

[12] Kolb, D., *Experiential Learning: Experience as the Source of Learning and Development*. Prentice-Hall Inc,1984.

[13] Lewin, K., ed. Resolving social conflicts; selected papers on group dynamics. ed. Lewin, G., W. 1948, Harper & Row.

[14] Makatchev, M., Jordan, P., and VanLehn, K., Abductive Theorem Proving for Analyzing Student Explanations and Guiding Feedback in Intelligent Tutoring Systems. *Journal of Automated Reasoning and Theorem Proving in Education*, 2004. **32**(3): p. 187-226.

[15] Nkambou, R., Frasson, C., and Gauthier, G., CREAM-Tools: An Authoring Environment for Knowledge Engineering in Intelligent Tutoring Systems. In *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software*, Murray, T., et al., (eds) 2003, Kluwer Academic Publisher.

[16] Nkambou, R. and Tchetagni, J. Diagnosing Learner Errors in a Web Based environments using the MPE. In Uskov, V. (*eds*), *Proceedings of WBE2004*. Acta Press. p. 135-145

[17] Person, N., et al., Simulating human tutor dialog moves in AutoTutor. *International Journal of Artificial Intelligence in Education*, 2001. **12**: p. 23-39.

[18] Peylo, C., et al. A Web-based intelligent educational system for PROLOG. In Gauthier, G., et al. (*eds*), *Proceedings of ITS2000*. p. 85-96

[19] Rosé, C. and Torrey, C. DReSDeN: Towards a Trainable Tutorial Dialogue Manager to Support Negotiation Dialogues for Learning and Reflection. In Lester, J., et al. (*eds*), *Proceedings of ITS2004*. Springer. p. 412-422

[20] Rosé, P., et al. Interactive Conceptual Tutoring in Atlas-Andes. In Moore, J.D., et al. (*eds*), *Proceedings of AIED2001*. IOS Press. p. 255-266

[21] Schön, D., A., *The reflective practitionner. How professionals think in action*. TempleSmith,1983.

[22] Sison, R., Numao, M., and Shimura, M., Multistrategy Discovery and Detection of Novice Programmer Errors. *Machine Learning*, 2000. **38**(1): p. 157-180.

[23] Tchetagni, J. and Nkambou, R. A framework for epistemological diagnosis and remediation in learning Logical Programming. In Pernin, J.-P., et al. (*eds*), *Proceedings of CALIE04*. University Joseph Fourier. p. 135-146

[24] Tchetagni, J., Nkambou, R., and Kabanza, F. Epistemological Remediation in Intelligent Tutoring Systems. In Orchard, B., et al. (*eds*), *Proceedings of IEA/AIE2004*. Springer-Verlag. p. 955-966

[25] Vassileva, J., McCalla, G., and Greer, J., Multi-Agent Multi-User Modelling in I-Help. *User Modelling and User Adapted Interaction*, 2003. **13**(1): p. 179-210.

[26] Zapata-Rivera, J.-D. and Greer, J. Analyzing Student Reflection in The Learning Game. In Hoppe, U., et al. (*eds*), *Proceedings of AIED2003*. IOS Press. p. 288-298

Modelling and Externalising Learners' Interaction Behaviour

Kyparisia A. Papanikolaou and Maria Grigoriadou
Department of Informatics & Telecommunications, University of Athens, Panepistimiopolis, GR 15784, Athens, Greece
spap@di.uoa.gr, gregor@di.uoa.gr

Abstract. In this paper, we propose an approach to modelling learner interaction behaviour by combining learners' interaction protocols with different variables that affect their selections, and interpret this model with the aim to characterise learners' activity. Moreover, we investigate the issue of how this information could be externalised to learners in order to support reflection and develop their awareness of style issues. An application example of the specific approach to the adaptive educational system INSPIRE is also provided.

1 Introduction

Adaptive Educational Systems (AESs) possess the ability to make intelligent decisions about the interactions that take place during learning and aim to support learners without being directive (Brusilovsky, 1996; Brusilovsky and Peylo, 2003). During the interaction an AES need to track learner usage and analyze his/her activity in order to dynamically adapt the content presentation, topic sequencing, navigation support. The type, use and focus of analysis can vary depending on what it is intended for. Moreover, an important characteristic of AESs is the sharing of control between the learner and the system, as several levels of adaptation can be distinguished depending on who takes the initiative: the learner or the system (Kay, 2001). Systems that integrate adaptive and adaptable components are based on shared decision making requiring shared knowledge between the learner and the system. Thus, if the system decides based on an analysis of learners' interaction then the learner should also be aware of this information in order to share the same knowledge and be able to decide. It would be worthwhile to investigate how an analysis of learner-system interaction could stimulate reflection on the learning process and enhance learner control opportunities.

Especially in educational systems that aim to support learners fulfil their learning goals, opportunity for reflection must be central. Several researchers have studied learner reflection using a variety of strategies to support interaction with the learner model (Bull et al., 2001; Morales et al., 2001; Dimitrova et al., 2001; Aleven and Koedinger, 2000; Zapata-Rivera and Greer, 2003). It has been argued that externalising representations of learners' understanding can raise their awareness of their knowledge, progress, difficulties and the learning process, which should in turn, lead to enhanced learning (Dimitrova et al., 2001; Bull and Nghiem, 2002; Mitrovic and Martin, 2002;). In the case of externalising learners' behaviour, as only a limited set of events from learners' interaction with an educational system is available, a challenging research goal is to investigate learner-demonstrated behaviour, relating learners' goals, selections, and navigation patterns. Furthermore, mirroring data about system usage to learners could be especially valuable allowing a direct observation of their learning strategies. Different issues that need to be addressed are how to collect, analyse and externalise data from learner interactions. Which specific measures of learners' observable behaviour are relevant indicators of learner studying preferences? Which characteristics of the resources that learners encounter during interaction, should be considered in an analysis of their navigation patterns? How can we 'describe' the context that affects learner selections and studying preferences, including tools and support offered to learners? How to interpret navigation patterns in a meaningful way to support reflection, which lead them to the accomplishment of their goals? These research goals may help us develop deeper knowledge of the complex interactions that are involved during a learning task between the learner and the system, and further inspire new approaches towards more learner-centred designs of adaptive educational systems.

In this paper, we attempt to model learner interaction behaviour combining learners' interaction protocols with different attributes of the instructional design of the system, and interpret this model with the aim to characterise learners' activity. Moreover, we investigate the externalisation and visualisation of this information in order to support learner reflection on their learning strategies and enhance awareness of style issues, which may lead them to improve their interaction. An application example of the specific approach to the adaptive educational system INSPIRE is also provided.

2 Investigating and externalising learners' observable behaviour

Modelling learner behaviour requires a thorough investigation of the cognitive processes at work during an interaction. Different research studies have investigated how these cognitive processes can be measured and linked with learner's social interaction or with specific characteristics of the learner such as the cognitive/learning styles. As far as AESs is concerned, in several systems learner's interaction with educational resources is used as the main source of adaptation. Lately, in a few AESs raw data from learner's interaction behaviour is externalised to learners.

Several studies have explored ways to characterise navigational path data investigating which measures of learners' observable behaviour are indicative of their learning behaviour. In a pilot study investigating if path data could be used as a basis for diagnosing differences in learning style (Andris & Stueber, 1994), different measures that were accounted were sequence of node, time at node, time per node, linearity, and reversibility. Reed et al. (2000) investigate the relationship of learners' computer experience and learning style on hypermedia navigation in terms of the linear/nonlinear steps, the percentage of nonlinear steps they performed and the time on task they spent. Ford and Chen (2000) have shown that people with different cognitive styles display different learning strategies when they are allowed to navigate in relatively non-linear learning environments. For example, there were differences in the subject categories and navigation tools selected, in the total number of navigational actions, in the numbers of levels visited and the time spent at each level, and in the sequencing of elements explored. Indicators that have been investigated for several learning/cognitive style categorizations are (Reed et al., 2000; Lu et al. 2003; Papanikolaou et al, 2003): (*i*) navigational indicators (sequence of node, linearity, reversibility); (*ii*) temporal indicators (time at node, time per node); (*iii*) performance indicators (total learner attempts on exercises, performance on tests).

Moreover, a restricted set of events from learners' interaction, such as requests on learning materials and progress as it depicted by the submission of tests, has been used in several AESs to guide adaptation during the interaction. ACE (Specht and Opperman, 1998) dynamically adapts the instructional strategy based on information coming from monitoring learner's requests on learning materials, as well as on the success of the currently used strategy; repeated occurrences of high performance in tests raise the preference value of a strategy until a threshold is reached. Also, Arthur (Gilbert and Han, 1999) dynamically adapts the instructional style according to learner's performance in the tests s/he submits. Lastly, MANIC (Stern and Woolf, 2000) uses machine learning techniques in order to identify learners' preferences by observing his/her interactions with the system. ELM-ART (Weber and Brusilovsky, 2001) and KnowledgeTree (Brusilovsky, 2004) keep data about learners' interaction with the system and externalise this information to learners in the form of statistics.

In the above studies specific aspects of learner's interaction behaviour are either used by the system as a source of adaptation or externalised to learners in a text-based form illustrating learners' performance results or final learning state. However, learners' observable behaviour during an interaction could be further exploited to provide a comprehensive view of learners' cognitive activity as it unfolds in addition to the activity outcomes, their preferences and progress in a particular context. To this end, a description of the interaction need to be developed involving the learner's actions with the educational content and system functionalities. Several research studies in the area of collaborative learning systems have explored the issue of analyzing learners' interaction in a social context and externalize these data to learners in several ways in order to enhance reflection and support collaborative interaction (Jerman et al., 2001). Moreover, as valuable resources in modeling and visualizing learners' interaction could be used research studies investigating learners' interaction with different stakeholders of the educational process such as teachers and resources (ASTILIEO, 2004).

3 Modelling learner's interaction: *guiding through mirroring*

In a learning environment learners have to make explicit decisions repeatedly during interaction. However, interaction protocols, referring to the series of events which occur during hypermedia usage with corresponding time stamps (Rouet and Passerault, 1999), are sometimes very complex including sets of heterogeneous data which must be carefully handled in order to yield meaningful information. In this process, key issues are the selection of the appropriate data, their interpretation, and the way this information is conveyed to the learner. The task of interpretation demands the construction of a model of the interaction, which is instantiated to represent the current state of interaction, and possibly the state of interaction proposed by the instructional design of the system or the state of interaction of peers. It is then up to the learner to interpret the visualization and decide what actions (if any) to take.

In particular, the process of modelling learner's interaction with the double aim of providing a mirror of the learner's actions and useful recommendations whenever a perturbation arises, contains the following phases:

- Phase 1. The data collection phase involves observing and recording the interaction. Specific *indicators* of learners' interaction with the content and functionalities of the system are logged and stored for later processing.
- Phase 2. This phase involves selecting one or more *attributes* of the instructional design of the system, which provide meaningful information about learners' actions and thus they are valuable in characterizing learner's interaction.
- Phase 3. In the final phase, *an analysis of the interaction* is provided by comparing the current state of interaction to the one proposed by the instructional design of the system and/or the one adopted by peers.

Phase 1. The study of learners' behaviour through interaction protocols demands the definition of relevant *indicators* from learners' interaction with the content and functionalities of the system, which will be further linked to specific characteristics of the learner in an analysis of the interaction. Moreover, an essential step in this study, is the definition of the appropriate *observation grain*, which relates to the precision of the events considered as units in the analysis of the interaction protocols. This relates to the events that need to be analysed, ranging from global activity patterns, which can be used to capture global features of subjects' representations or strategies (coarse grain) when studying a goal, to specific aspects of the interaction (intermediate or fine grain) (Rouet and

Passerault, 1999). The observation grain links to the study objectives and in our case a combination of different levels of grain seems appropriate.

As indicators of learners' interaction with the content and functionalities of the system that link to learners' cognitive activity are considered specific *navigational* and *temporal* indicators such as visits on resources, sequencing of resources, time spent on the resources. Moreover, as indicators of learner's performance are used specific *performance indicators* showing learners' success with resources, such as total attempts on assessment questions, performance in tests, etc. In this phase, quantitative and qualitative data at different levels of observation grain, ranging from global activity to specific aspects of the interaction, should be used and shaped to the individual characteristics of the educational system and the study objectives. For example, the coarse grain approach can be used to capture global features of the learner's state such as the different topics s/he worked with, his/her knowledge level on these topics. However, this observation grain does not permit to answer more specific questions about learner's study preferences and strategies when working with specific topics. At the intermediate grain approach analysing the interaction protocol amounts to selecting the events of interest and making appropriate computations (e.g. frequency). This approach is useful when testing specific hypotheses about the cognitive processes at work during the interaction. For example, in case of investigating learners' preferences of resources for a specific topic, then the frequency of learner's selections of various educational material types - e.g. examples, exercises, theory presentations, activities - is of great importance. Lastly, at the fine grain level, all the observable actions are taken into account. In this approach the complete sequence of events included in raw interaction protocols are analysed. In this case the investigator focuses on meaningful patterns, in order to achieve an understanding of learner's activity when working with particular tasks. Data selection or interpretation requires a careful analysis of the task and study objectives. For example, the analysis of navigation patterns to examine the way learners use assessment may result to different strategies such as using tests as a self-assessment tool during studying or using assessment questions to guide the learner's study in particular topics.

Phase 2. An analysis of the learner's interaction protocols, in order to provide meaningful information about learners' cognitive activity, should take into consideration different *attributes* of the instructional design of the system. In the proposed approach, as such *attributes* could be considered (a) the type, semantic density, view of the *resources* that the learner encountered during interaction, and (b) the type and functionality of tools selected, the support (navigational aids, sequencing or presentation of content) exploited by the learner to accomplish his/her goals and tasks (*context*).

Phase 3. The system visualises all the above data providing the current state of the learner (mirroring) aside the state of interaction proposed by the instructional design of the system or the state of interaction of other peers. Moreover, the system analyses all the data collected in the previous phases and make *recommendations* to the learner about how to proceed. The analysis of learners' actions is based on his/her interaction with the *resources* in the current *context*.

The main idea behind such a learner modelling approach is to (*i*) enable mirroring of learners' interaction along with an interaction state derived form the instructional design of the system or adopted by other learners, with the aim to support reflection which may lead to successful behaviour, (*ii*) support the provision of meaningful recommendations to learners, (*iii*) guide system adaptive behaviour in case of an agreement between the learner and the system.

4 An application example

INSPIRE (Papanikolaou et al., 2003) is a web-based Adaptive Educational Hypermedia system designed to support web-based instruction. In INSPIRE, learners have always the option to select and study the learning goal they prefer independently of their previous selections; all the material necessary for their study is provided when a learning goal is selected. In particular, INSPIRE plans the content of instruction for the particular learning goal, i.e. selects the contents of a sequence of lessons that gradually support learners to achieve their goal. INSPIRE aims to facilitate learners during their study, providing personalized instruction (a) proposes a navigation route through the lesson contents based on learner’s knowledge level and progress, and (b) adapts the presentation of the educational material to the learners' learning style.

The educational content of INSPIRE is represented in three hierarchical levels: *learning goals, concepts and educational material* (see Figure 1). A learning goal corresponds to a topic of the domain knowledge. Each goal is associated with a subset of concepts of the domain, which formulates a conceptual structure that represents all the concepts of the goal and their relationships (outcomes, prerequisites, related). Each outcome concept of a goal is associated with several educational material pages which consist of *knowledge modules* such as theory presentations, questions introducing or assessing the concept, examples, exercises, activities. The educational material pages of each outcome are organised in three level of performance (see Figure 1): (i) Pages of the *Remember* level include knowledge modules that introduce the concept and make learners speculate on newly introduced ideas, such as theory presentations, questions (introductory or self-assessment), and examples, (ii) Pages of the *Use* level of performance include knowledge modules that support learners to apply the concept to specific case(s), such as hints of the theory, examples, exercises, activities, (iii) Pages of the *Find* level of performance include knowledge modules necessary to stimulate learners to find a new generality, principle, procedure, through specific activities. All learners receive the same knowledge modules. However, the order and mode (embedded or link) of their presentation in a page is adapted based on the learner’s learning style – the (Honey and Mumford, 1992) categorisation has been adopted which suggests four types of learner: *Activist, Pragmatist, Reflector, Theorist.*

It should be emphasised that learners working with INSPIRE have the option to select the educational material they prefer to study, in the order they prefer. In particular, INSPIRE supports several levels of adaptation from purely adaptive to purely adaptable. Learners working with INSPIRE have the options (a) to follow recommendations proposed by the system (b) to access their learner model, reflect upon its contents and change them in order to guide system’s instructional decisions (see Fig.1 - Learner Model), (b) to deactivate the dynamic lesson generation process and select the next lesson contents (see Fig.1 - Lesson tool). In the current version of the system the learner model provides information about the knowledge level of the learner on the outcome concepts of the current lesson, as well as about his/her learning style. The design approach presented below is the first step towards the extension of the learner model of INSPIRE to include information about learner’s interaction behaviour. This new version is currently under development.

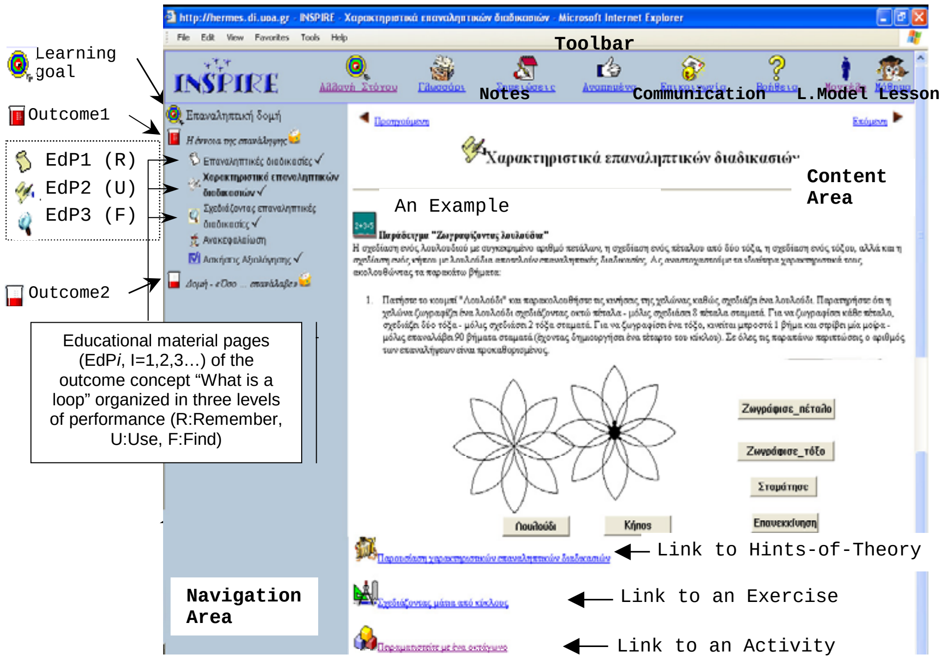


Figure 1. The main screen of INSPIRE (http://hermes.di.uoa.gr/inspire) provides learners with a complete view of the structure of the domain knowledge (Navigation Area) and direct access to learning resources (Content Area) and systems' functionalities (Toolbar). The lesson contents for the particular learner who studies a learning goal on Computer Programming, include two outcome concepts (Navigation Area): Outcome1 and Outcome2. Different icons are associated with the two outcomes, denoting variations of learner's knowledge level on the corresponding concepts: the learner has mastered Outcome1 (an almost full measuring cup appears next to Outcome1) whilst his/her knowledge level has been evaluated as {Mediocre} with regards to Outcome2 (a half empty measuring cup appears next to Outcome2). In the Navigation Area, Outcome1 has been expanded (only one outcome concept can be expanded at each time). Different icons are associated with the educational material pages (EdP*i*) of Outcome1 denoting the level of performance to which each page corresponds (R:Remember, U:Use, F:Find). At the Content Area a page of the Use level (as it appears to a Reflector) appears which includes the knowledge modules: *Example*; link to *Hints-of-Theory*; link to an *Exercise;* link to an *Activity.*

Modelling learner's observable behaviour. In this paragraph we present how the modelling approach described in Section 3 will inform the design and externalisation of the learner model of INSPIRE. The system gathers data from learner's interaction with the system and shows some visualisations of this information in order to support learners gather evidence to evaluate the efficacy of their moves. To this end the visualisations include a set of indicators that represent the state of interaction along with a set of desired values for those indicators. These values may be derived from the instructional design of the system or may reflect their peers' interaction. As will be described below, the values derived from the instructional design of the system refer to (a) the semantic density of the resources as proposed by the tutor, e.g. the time that the learner has spent on specific resources, will be presented aside the semantic density of the resources, as well as (b) the navigational advice offered based on the learner's knowledge level, e.g. the resources that the learner selects will be presented aside the system's navigational advice. In particular, through mirroring learners' interaction behaviour we aim to (a) support learners' reflection on their strategies and in cases of failure guide them towards successful behaviour; (b) enhance learners' style awareness; (c) guide system adaptive behaviour; (d) support tutors in providing personalised guidance and instruction and evaluate the educational resources.

In particular, we use navigational (number of hits, frequency of visits, sequence of first visit/revisits), temporal (time spent on different types of resources and assessment), and performance (attempts on assessment questions, performance on tests) *indicators* of learner's interaction which are recorded at three levels of grain, coarse, intermediate, fine, in order to provide a comprehensive view of learner's cognitive activity (see Table I). In a pilot study we performed to investigate learners' beliefs about the usefulness of such information, revealed that most learners believe that this type of data reflects their study preferences and that most of them would use these data to improve their interaction behaviour in case of failure. Moreover the above indicators are linked to different *attributes* of the instructional design of INSPIRE that provide useful information about learners' actions (a) semantic information of the resources such as type (theory presentations, examples, assessment, activities), view (Activist's, Reflector's, Theorist's, Pragmatist's), semantic density (study time proposed by tutor) and (b) context information including the tools (see Fig. 1 the tools: *Notes*, *Communication* that includes chat, discussion lists, e-mail, *Learner Model* through which learners may check and update their knowledge level and learning style, Lesson through which learners may deactivate the adaptive behaviour of the system) and support (navigation advice) offered. Lastly, three different levels of observation grain have been considered, following the structure of the domain knowledge (goals, concepts, educational material pages/modules), each one having a different study objective: (a) at the *coarse level*, information about the general activity of the learner during studying a specific learning goal is provided. This information provides a means to evaluate the learner's involvement in the goal through the total time s/he spent on the goal (at the current session, total study time) as well as the learner's activity such as which outcomes s/he has studied, his/her performance, etc.; (b) at the *intermediate level*, information about the learner's cognitive activity on each particular outcome concept of the goal is provided. This includes time spent on different types of resources (educational material pages, knowledge modules), time spent/frequency of visits on specific types of resources/activities, information about assessment (time spent, number of questions answered, attempts, performance), etc. The information at this grain illustrates the way the learner uses different types of resources –which is linked to learner's style preferences - and the impact on her performance. Moreover, information about the time spent on specific resources combined with the semantic density of the resources and the learners' knowledge level could provide a means to evaluate the difficulties that a learner faces with the educational material as well as the quality and adequacy of specific resources for learners with a particular knowledge level; (c) at the *fine grain* information about the learner's cognitive activity on particular tasks is provided. Learner's navigation patterns are recorded including information about the learner's actions/selections of resources and specific tools, aside with system advice. The information at this grain allows the investigation of the evolution of learner's activity over time and how this relates to his/her style preferences and knowledge level. For example, information about the resources the learner uses when working with activities or solving an exercise (e.g. the sequencing of resources s/he adopts) linked with his/her performance could illustrate his/her learning strategies as well as their success or failure. The investigation of repetitive patterns of learners' behaviour and the way these patterns link to the learner's style preferences, knowledge level and performance could provide a deeper view on the way learners use the resources and interact with the educational system.

In Figures 2 and 3 we provide two mock-ups of the visualization of learners' interaction behaviour at the coarse and intermediate observation grains, as these will appear in the learner model. Figure 2 represents data about learner's interaction at the coarse observation grain, i.e. during studying a learning goal, whilst Figure 3 at the intermediate observation grain, i.e. during studying a specific outcome concept of the learning goal. For a detailed description of the interaction behavior data visualized in both Figures 2, 3 see Table I. In both Figures the black lines denote the current state of the learner's interaction whilst the double ones, the state proposed by the instructional design of the system.

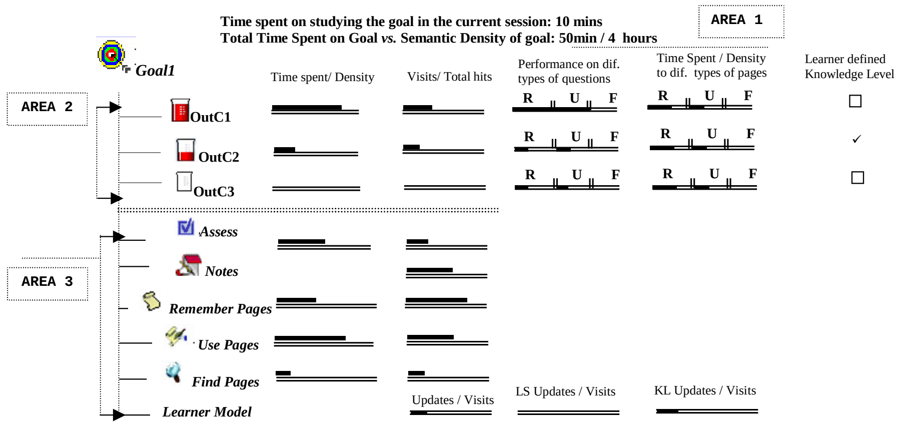


Figure 2. A mock-up of the visualization of learners' interaction with the resources and system functionalities at the coarse observation grain. Note that the different icons that accompany the educational material pages and the outcome concepts reflect the type of resources and the knowledge level of the learner (see also Figure 1). Note that: (a) the outcomes that the learner has to study are OutC*i*: outcome concept *i*=1,2,3; (b) the different levels of performance on which the educational material pages of the outcomes correspond are: R:Remember, U:Use, F:Find; (c) the different functionalities the learner used are *Assess*: submission of assessment tests, *Notes*: answering on questions, exercises, activities using the Notes tool from the toolbar, *Learner Model*: visits/updates on the learner model which presents the learners' Knowledge level (KL) and Learning Style (LS). For a detailed description of the interaction behavior data visualized in this figure see Table I – *Coarse grain*.

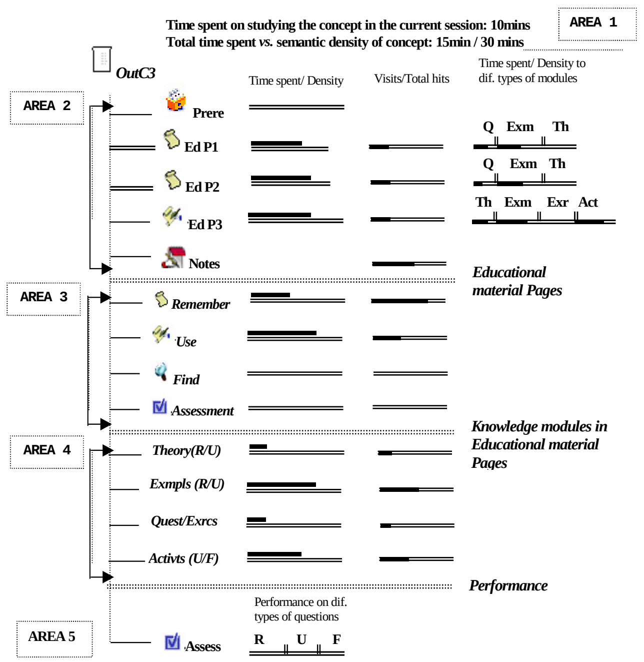


Figure 3. A mock-up of the visualization of learners' interaction with the resources of the outcome concept "OutC3" of the learning goal (intermediate observation grain). Note that: (a) the educational material pages are: Prere which present the prerequisite concepts of the outcome, Ed P*i*: educational material pages of the OutC3; (b) the different knowledge modules included in the educational material pages are Q: Question, Exm: Example, Th: Theory, Exr: Exercise, Act: Activity; (c) the different levels of performance on which the educational material pages correspond are: R: Remember, U: Use, F: Find. For a detailed description of the interaction behavior data visualized in this figure see Table I - *Intermediate grain*.

Table I. Data describing learner's interaction behaviour at three observation grains: coarse, intermediate, fine.

Coarse grain (Goal level)
Data visualized in AREA 1 of Figure 2
- Time spent on studying the goal in the current session (see Figure 2 – the goal is *Goal1*)
- Total time spent on studying the goal *vs.* semantic density of goal resources
Data visualized in AREA 2 of Figure 2
The following information is provided for each outcome concept of the goal (see Figure 2 – OutCi):
- Time spent on studying a concept of the goal *vs.* semantic density of the concept resources
- Frequency of visits: Visits on the resources of a concept of the goal *vs.* total hits on concept resources
- The learner's correct answers on the assessment test of a concept of the goal analyzed in her performance on different types of questions *vs.* total number of different types of questions (R: Remember, U: Use, F: Find)

- Time spent on studying the educational material pages of the (Remember, Use, Find) level of a concept of the goal *vs.* semantic density of the particular resources
- Information about how the learner's Knowledge Level (KL) was estimated: through tests or defined by the learner

Data visualized in AREA 3 of Figure 2

- Time spent on assessment *vs.* semantic density of the assessment tests
- Frequency of visits: Visits on the assessment tests of a goal *vs.* total hits on goal resources
- Frequency of visits: Visits on the Notes tool (where learners submit their answers to questions, exercises, activities) *vs.* total hits on questions, exercises, activities of the goal
- Time spent on studying the educational material pages of the (Remember, Use, Find) level of performance *vs.* semantic density of the particular resources of the goal
- Frequency of visits: Visits on the educational material pages of the (Remember, Use, Find) level *vs.* total hits on the educational material pages of the goal
- Updates of the learner model *vs.* visits on the learner model plus updates of Learning Style (LS) information *vs.* total visits on the model plus updates of Knowledge Level (KL) information *vs.* total visits on the model

Intermediate grain (Concept level)

Data visualized in AREA 1 of Figure 3

- Time spent on studying the concept in the current session (see Figure 3 – the concept is *OutC3*)
- Time spent on studying the concept *vs.* semantic density of concept resources

Data visualized in AREA 2 of Figure 3

- Time spent on studying prerequisite concepts of the outcome *vs.* semantic density of the particular resources
- Time spent on studying the educational material pages of the (Remember, Use, Find) level of the concept *vs.* semantic density of the particular resources
- Frequency of visits: Visits on the educational material pages of the (Remember, Use, Find) level of the concept *vs.* total hits of the learner on the educational material pages of the concept
- Time spent on studying the knowledge modules (Question, Example, Exercise, Theory, Activity, Hints of Theory) included in educational material pages of the (Remember, Use, Find) levels of the concept *vs.* semantic density of the particular resources
- Frequency of visits: Visits on the Notes tool to submit their answers to questions, exercises, activities *vs.* total hits on questions, exercises, activities of the concept

Data visualized in AREA 3 of Figure 3

- Time spent on studying the educational material pages of the (Remember, Use, Find) level of performance and the assessment test page *vs.* semantic density of the particular resources
- Frequency of visits: Visits on the educational material pages of the (Remember, Use, Find) level and the assessment test page *vs.* total hits on the concept resources

Data visualized in AREA 4 of Figure 3

- Time spent on studying the knowledge modules (Question, Example, Exercise, Theory, Activity, Hints of Theory) included in educational material pages of the (Remember, Use, Find) levels of a concept *vs.* semantic density of particular resources
- Frequency of visits: Visits on the knowledge modules (Theory at the Remember (R) and Use (U) levels, Examples at the R/U levels, Questions/Exercises, Activities at the Remember and Find levels) included in pages of the (Remember, Use, Find) levels of a concept *vs.* total hits on the knowledge modules of the concept

Data visualized in AREA 5 of Figure 3

- The learner's correct answers on the assessment test of the concept analyzed in her performance on different types of questions *vs.* total number of different types of questions (R: Remember, U: Use, F: Find)

Fine grain (Educational material level)

- Learner's navigation pattern on the knowledge modules included in educational material pages of the (Remember, Use, Find) levels and assessment pages at a goal level (first time visit)
- Learner's navigation pattern on the knowledge modules included in educational material pages of the (Remember, Use, Find) levels and assessment pages at each specific outcome concept of a goal
- History of learners' selections of pages (including information about the concept, the level of page, time spent, system proposals/advice), visits/updates on the Learner Model (including model view, options changed), assessment (total attempts and performance), visits/actions with the Lesson, Notes, Communication tools.

5 Conclusions and Future Plans

In this paper we propose an approach to model learner interaction based on learner-system interaction protocols. Interaction protocols provide direct access to learners' activity rather its outcomes and thus they can be used as dependent measures to understand the nature of the learning process. In the proposed approach learners' interaction behaviour is modelled based on a combination of analysis of learner's actions with quantitative indicators such as time spent on resources, frequency of re-visits on specific resources, etc. We also investigated the externalisation of this information to the learner through the learner model in a meaningful way with the aim to support reflection and awareness of style issues, which may lead to improve their interaction and accomplish their goals.

Currently the new version of INSPIRE is under development. The proposed approach of modelling and externalising learners' interaction through the learner model will be further evaluated. Based on these results as well as on data about learners' interaction with their model we aim to design a negotiated model that will allow the learner and the system to jointly discuss the contents of the model. Moreover, in the near future we intend to further investigate (a) the visualisation of learner's navigation patterns at the fine observation grain, (b) the recommendations provided by the system, and (c) the selection and visualisation of data about peers' interaction behaviour augmented with comments about successful and unsuccessful interactions.

References

1. Aleven, V., & Koedinger, K.R. (2000) The Need For Tutorial Dialog To Support Self-Explanation. In C.P.Rose & R. Freedman (Eds.), Building Dialogue Systems for Tutorial Applications, AAAI Fall Symposium. 65-73.
2. Andris, J. (1996). The relationship of indices of student navigational patterns in a hypermedia geology lab simulation to two measures of learning style. *Journal of Educational Multimedia and Hypermedia*, 5, 303–315.
3. ASTILIEO (2004): Proceeding of the Workshop on Analyzing Student-Tutor Interaction Logs to Improve Educational Outcomes held at ITS2004. Available at: http://www.itsconference.com.br/content/workshops.html
4. Brusilovsky, P. (1996). Methods and Techniques of Adaptive Hypermedia. *User Modeling and User-Adapted Interaction* 6 (2/3), 87-129.
5. Brusilovsky, P. (2004). KnowledgeTree: A distributed architecture for adaptive e-learning, *in Proceedings of The Thirteenth International World Wide Web Conference*, WWW 2004, New York: ACM Press, 104-113
6. Brusilovsky, P. and Peylo, C. (2003). Adaptive and Intelligent Web-based Educational Systems. International Journal of Artificial Intelligence in Education 13 156–169.
7. Bull, S. & Nghiem, T. (2002). Helping Learners to Understand Themselves with a Learner Model Open to Students, Peers and Instructors, in P. Brna & V. Dimitrova (eds), *Proceedings of Workshop on Individual and Group Modelling Methods that Help Learners Understand Themselves*, International Conference on ITS 2002, 5-13.
8. Bull, S., Brna, P., Critchley, S, Davie, K, & Holzherr, C. (1999) The Missing Peer, Artificial Peers and the Enhancement of Human-Human Collaborative Student Modelling. *In proceedings of AIED 99*. pp. 269-276.
9. Dimitrova, V., Self, J. & Brna, P. (2001). Applying Interactive Open Learner Models to Learning Technical Terminology, in M. Bauer, P.J. Gmytrasiewicz & J. Vassileva (eds), *User Modeling 2001: 8th International Conference*, Springer-Verlag, Berlin Heidelberg, 148-157.
10. Ford, N., and Chen, S. Y. (2000). Individual differences, hypermedia navigation, and learning: An empirical study. *Journal of Educational Multimedia and Hypermedia*, 9, 281–312.
11. Gilbert, J.E. and Han, C.Y.: Adapting instruction in search of 'a significant difference. Journal of Network and Computer Applications 22 (1999) 149-160.
12. Honey, P. & Mumford, A. (1992). *The Manual of Learning Styles*, P. Honey, Maidenhead.
13. Jerman, P., Soller, A. and Muhlenbrock, M. (2001). From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning, *in P. Dillenbourg, A. Eurelings, and K. Hakkarainen (eds.), Proceedings of EuroCSCL*, 324-331, Maastrict, the Netherlands.
14. Kay, J. (2001). Learner control. *User Modeling and User-Adapted Interaction* 11(1/2) 111-127.
15. Lu, J., Yu, C.-S., Liu, C. (2003). Learning style, learning patterns, and learning performance in a WebCT-based MIS course. *Information & Management* 497-507.
16. Mitrovic, A. and Martin, B. (2002). Evaluating the Effects of Open Student Models on Learning, in P. De Bra, P. Brusilovsky and R. Conejo (eds), *Adaptive Hypermedia and Adaptive Web-Based Systems, Proceedings of Second International Conference,* Springer-Verlag, Berlin Heidelberg, 296-305.
17. Morales, R., Pain, H., and Conlon, T. (2001). Effects of Inspecting Learner Models on Learners' Abilities, *Proceedings of AIED'01*. IOS Press, pp. 434-445.
18. Papanikolaou, K.A., Grigoriadou, M., Kornilakis, H., Magoulas, G.D. (2003). Personalizing the interaction in a Web-based educational hypermedia system: the case of INSPIRE. *User-Modeling and User-Adapted Interaction* 13 (3) 213-267.
19. Reed, W.M., Oughton, J.M., Ayersman, D.J., Ervin, J.R., Giessler, S.F. (2000). Computer experience, learning style & hypermedia navigation. *Computers in Human Behavior* 16 609-628.
20. Rouet, J-F., and Passerault, J.-M. (1999). Analysing learner-hypermedia interaction: An overview of on-line methods. *Instructional Science* 27 201-219.
21. Specht, M. and Opperman, R.: ACE-Adaptive Courseware Environment. The New Review of Hypermeida and Multimedia 4 (1998) 141–161.
22. Stern, M.K. and Woolf, B.P.: Adaptive Content in an Online Lecture System. In: Brusilovsky, P., Stock, O., Strapparava, C.: Adaptive Hypermedia and Adaptive Web-based Systems, LNCS 1892 (2000). 227-238
23. Weber, G. & Brusilovsky, P. (2001). ELM-ART: An Adaptive Versatile System for Web-Based Instruction, *International Journal of AI in Education* 12(4), 351-384.
24. Zapata-Rivera, J.D. and Greer, J. (2003). Analyzing Student Reflection in The Learning Game. AIED'03 workshop on Learner Modelling for Reflection. 288-298.

Opening a Fuzzy-Learner Model

Selvarajah MOHANARAJAH, Ray KEMP, Elizabeth KEMP
Institute of Information Sciences and Technology
Massey University
New Zealand
{ S.Mohanarajah, R.Kemp, E.Kemp }@massey.ac.nz

Abstract: Learner models play important role in adaptable CBL systems. Opening Learner Models to the learners is considered beneficial for many reasons. However, opening Learner Models that use complex mathematical concepts like Bayesian Networks for numerical uncertainty handling is considered challenging. In contrast, explaining a fuzzy logic based Learner Model, as the variables resemble real world entities, is easier and does not require complex additional visualization systems.

This paper describes on-going research regarding opening up a fuzzy logic based student model. Moreover, we also discuss opening the underlying instructional strategies used in a Computer Based Learning (CBL) system that uses scaffolding techniques for mentoring.
Keywords: Pedagogical Action Sequencing (PAS), Opening Learner Models (LM).

1. Introduction

Maintaining optimal learner models has been considered challenging, mainly due to uncertainty associated with assigning credits between various probable causes based on observed evidences. Initially, the idea of opening Learner Models to the learners is suggested by Self [1], in order to share the burden in designing and maintaining adaptive Learner Models. However, the current trend in the research towards opening learner models reflects the contemporary learning theories. The constructivist and situated learning theories suggest that the instructional design should include strategies for enhancing collaborative learning and meta-cognitive activities such as reflection. Furthermore, opening up limited part of Learner Models to peers, and mentors also been given much attention in the recent research. This facilitate learners not only to estimate their own abilities but also compare it with their peers, Moreover, by giving facilities for and to express their views and concern to the peers and mentors, the act of opening Learner Model tries to mend the broken link between mentors and learners created by automation.

In our CBL system (called LOZ) for learning Object-Z notation[2], we use a formal approach to reduce the impact of uncertainty on pedagogical actions such as feedback. In particular, we use scaffolding techniques for mentoring (see [3, 4]). In this paper, which is an extension of [5], we discuss why our fuzzy approach in Learner Model is easy for systematically revealing not only various measurements kept by the system about the learner but also the methodologies (for uncertainty handling) used by the system for pedagogical action selections (PAS). Learners can also verify how various interactions with the system affect those measurements. Learners will find it easy to alter the measurement kept by the system for their ability in certain concept by just selecting their expected level of pedagogical actions retrospectively. We also discuss the issues related with opening the underlying instructional strategies (scaffolding processes) to the interested learners.

2. Opening Learner Models

The Learner Model includes abstraction of the beliefs of the system based on certain characteristics of the learner. Opening the learner model is an act of providing the information kept in a learner model (and how it would be utilized) in a usable format for other agents [6]. Self introduced the idea of opening the student model to the learners [1]. He and several others [6-10] have discussed various researches involved with opening Learner Models.

The following are some of the advantageous claimed for opening Learner Models;

(1) Reduces the burden of learner models significantly.
Maintaining optimal Learner Models are very difficult, if not impossible. Learners are allowed to change the measurements kept by the system. The role is now shared between the system and the learner. The possible misinterpretation by the system may be corrected by the learner at the early stages.

(2) Improves the learner's meta-cognitive activities.
Learners can view what they learned and how much they learned. They can backtrack how they learned and how they responded in various system states and for different system behavior. Learners could,
   a. reflect on their own knowledge and skills
   b. understand the weak areas and misconception
   c. estimate their own general qualities and capabilities
   d. identify their learning styles, preferences, etc.
   e. take control over their learning process

(3) Maintains the learner's intrinsic motivation
Learners can evaluate their performance over time. They can also view and compare peer's models (limited by law and ethics). Cognitive motivational theories such as self-efficacy theory suggests that the learners intrinsic motivation (by especially mastery and vicarious experience[11]) is increased

(4) Keeps Learner- Mentor link active
Mentors can view the learners' models individually and collectively. Learners could comment on their own learner model and other expectations. Mentor can motivate learners depending on their individual needs through different ways.

(5) Abides to the Laws in some countries.
Some countries have intensive privacy laws, which demand all the information kept by a system about a person should be revealed to the person, if requested.

Paiva et al [6] describe an open system called TAGUS and mentioned several other related research such as [12] . TAGUS separates the learner model and its functionality from the underlying applications. Mitrovic et al [9] discuss various other research, and particularly, present the encouraging results of the evaluation on one of their systems (SQL Tutor). Susan [7] in her keynote-address discussed variety of strategies and interfaces used in opening learner models with illustrations.

Importantly, in other research carried out in the ARIES lab [10], a graphical tool called VisiMod (Visualization of Bayesian Student Models) is used to assist the learners and teachers to inspect and modify a complex learner model that uses a Bayesian Network.

The parameters are mapped to fuzzy-like attributes (good, very-good, expert) and can be modified simply by using sliders. Two artificial agents are also used to guide the learners through the interaction. Extensive evaluations based on usability and explorative studies are also included in this paper.

Our research is distinct from that discussed above in the sense that we not only reveal the learner model and learning pattern of a student but also the strategies used in modeling and the scaffolding process in different phases. The gifted learners may speed-up or totally bypass the scaffolding process. In addition to that, as the measures used in the student model are all fuzzy variables, the real world meanings for these numerical figures can be easily illustrated. The graphical interface allows one to systematically modify the relevant measures easily and observe the effects of the proposed changes (will be discussed in section-4 in more detail).

3. Background: Domain and Learner Models for LOZ

The learning material associated with the domain knowledge in LOZ is organized into several sub-concepts [13]. Each sub-concept is associated with a series of mental states. We assume that a person is in a certain 'mental state' if they have already constructed sufficient knowledge (and/or skills) to perform a certain mental or physical task. Each mental state in the series represents a scaffolding stage. After presenting the relevant material for learning a concept, learners are guided through a learning path, gradually from the basic mental state to the highest mental state, using MCQs (Multiple Choice Questions) and relevant feed-back (in other words, from the first scaffolding stage to the final stage, one step at a time).

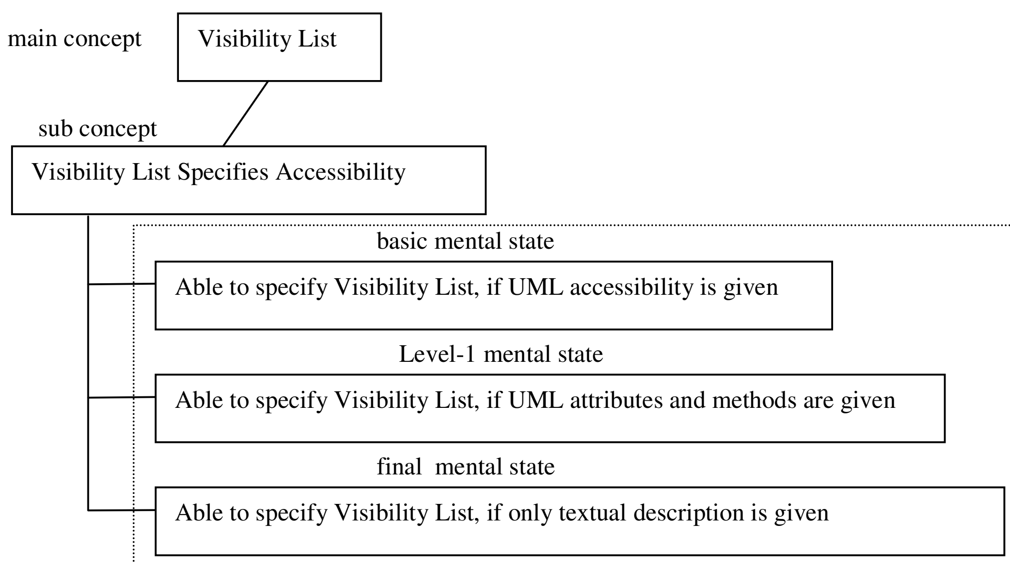


Figure 1. Example for Concepts and Mental States in LOZ

Figure 1 illustrates a portion of the domain model of LOZ. MCQs plays important role in our system; they are used as scaffolding blocks rather than assessment units. MCQs are carefully selected to match the relevant mental states. The inappropriate answers for each MCQ reflect some misconceptions associated with those mental states. The feedbacks are chosen to address those misconceptions. Initially, learners are given a UML specification and asked to produce a corresponding Object-Z specification. In this way, learners are initially freed from performing abstraction and allowed to concentrate on understanding the underlying mathematical concepts. Gradually, the UML help will be withdrawn. Eventually, the learner will be able to perform abstractions as well as be able to apply mathematical notations to represent those abstractions.

Level	PASw - for Wrong answer	PASc- for Correct answer
L1	Let them Answer-Once-Again (same MCQ) If answered correct in second time, give another MCQ in next Scaffolding Level (Move to next level)	Affirm (just inform that the answer is correct), Move to next level
L2	Explain Why the selected answer is Wrong , Let them Answer-Once-Again (same MCQ) If answered correct in second time give another MCQ, but in same Scaffolding Level (Stay in same level)	Explain why the selected answer is correct , Move to next level
L3	Explain why the selected answer is Wrong & why the system's answer is correct , Give another MCQ, but in same Scaffolding Level	Explain why the selected answer is correct & why other answers are wrong, Move to next level
L4	Explain why the selected answer is Wrong & why the system's answer is correct , Compare with Z and UML, if applicable give another MCQ, but in same Scaffolding Level	Explain why the selected answer is correct & why other answers are wrong, Provide topic related explanation Move to next level
L5	Explain why the selected answer is Wrong & why the system's answer is correct , Provide topic related explanation Compare with Z and UML, if applicable Give another MCQ, but in the next lowest Scaffolding level (Move to next-low level)	Explain why the selected answer is correct & why other answers are wrong, Provide topic related explanation Stay in same level

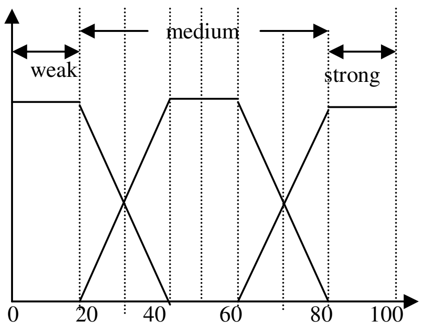
Figure 2. Different Levels of Pedagogical Action Selection

After learning a concept, the learner will be given an MCQ at its basic mental state (scaffolding level 1). Depending on the answer, the system should provide suitable pedagogical actions tailored to the learners (feedback and next action- whether to move to the next scaffolding level or not).

Feedback research has a long history. Partially based on Mason et al's [14] research, we designed two disjoint sets of fuzzy variables for the strength of pedagogical actions (figure 2), one for correct answer (PASc) and the other for wrong answer (PASw). Both variables PasC and PasW vary from one to five levels with increasing pedagogical strength. For example, the PAS-index 5 for correct answer (PAScL5) indicates that the system assumes the given MCQ is very hard for the particular learner; therefore needs some informative feedback even they answered it correctly (it may be a lucky guess). However, the learner will be given a problem in the same scaffolding level to confirm that the previous success is not a lucky guess. Whereas, the PAS-index 5 for wrong answer (PAScL5) indicates, in addition to giving a detailed feed back, the learner will be encouraged to get back to the previous scaffolding level.

To incorporate dynamic adaptability, the Learner Model in LOZ keeps some measures including Strength on Mental States (SMS). SMS indicates a person's strength on a particular mental state (related to a scaffolding stage). It cannot take yes-or-no type of values. Therefore, we let SMS be fuzzy [15] and take three values: Strong, Medium and Weak.

(a)



(b)

Mental State (SMS)	Difficulty level (D)	PASw Answer Wrong	PASc Answer Correct
Strong	Low	PASwL1	PAScL1
Strong	Moderate	PASwL2	PAScL2
Strong	High	PASwL3	PAScL3
Medium	Low	PASwL2	PAScL2
Medium	Moderate	PASwL3	PAScL3
Medium	High	PASwL4	PAScL4
Weak	Low	PASwL3	PAScL3
Weak	Moderate	PASwL4	PAScL4
Weak	High	PASwL5	PAScL5

Figure 3 (a). Membership Function for SMS and D **(b).** Fuzzy Rules for PASw and PASc

The fuzzy membership function of SMS we use in this research is given in figure 3a (a score SMS=75 means, the system believes 75% that the learner is strong and 25% medium in a particular mental state). For example, assume that SMS will be 50 (medium-100%) for the basic mental states of each concept (for more detail on the Learner Model of LOZ see [13]. Another fuzzy variable D, which takes three values: High, Moderate and Easy, represents the difficulty level of an MCQ. Note that the difficulty level of MCQs will closely match with their scaffolding level. We assume the same fuzzy membership function given in figure 3a for D also. After an MCQ is answered, the system uses some fuzzy rules (based on three variables; SMS, D and P-correct/wrong) to provide suitable pedagogical actions (figure 3b). SMS will be continuously updated against new evidence using Bayesian rules.

4. Opening the Learner Model in LOZ

In line with the Cognitive Apprenticeship model [4], we have used a four-phase instructional model for LOZ. However, our model does not include explicit phases for reflection (or articulation), instead, the system provides various facilities for learners to view and modify certain aspects of learner model and scaffolding strategies. During learning, learners can inspect their learning process, compare and discuss their performance with peers and mentor, and check the reasons behind various decisions made by the system, and finally, alter the learner model and scaffolding processes.

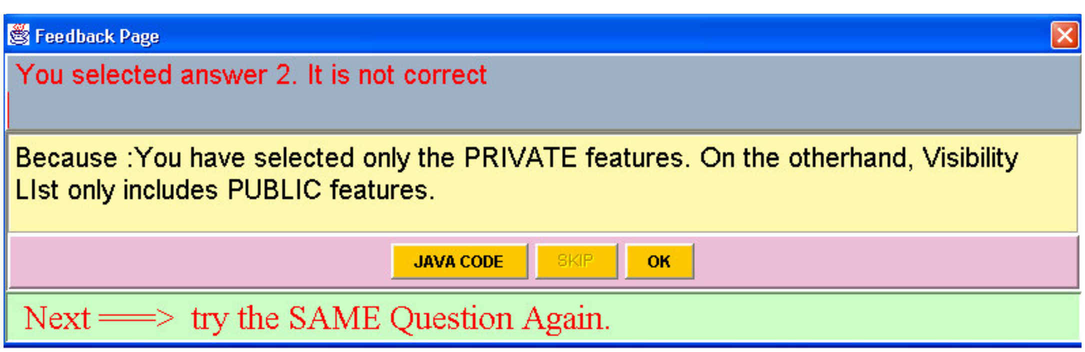
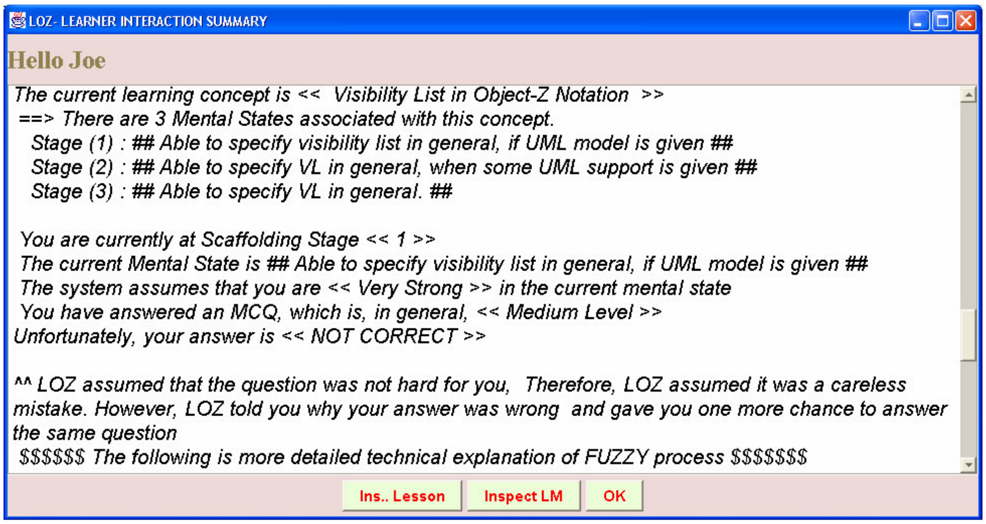


Figure 4: Feedback window for PASwL2

If a learner feels that the system is not providing sufficient feedback, probably the system ranked the learner higher than they should be. For example, if a learner with higher SMS score (say 90%) failed in a medium difficulty level question (say 55%), the system assumes it as a slip, and according to the fuzzy rules given in figure 3b, the level 2 PAS action for wrong answer (PASwL2) action will be selected. The PASwL2 action will just inform the user why the selected answer was wrong (figure 2) and give another opportunity to correct the error (figure 4). In this way, the system avoids overloading learners with unnecessarily detailed feedback and forcing them to stick on a lower level scaffolding stage.

(a):



(b):

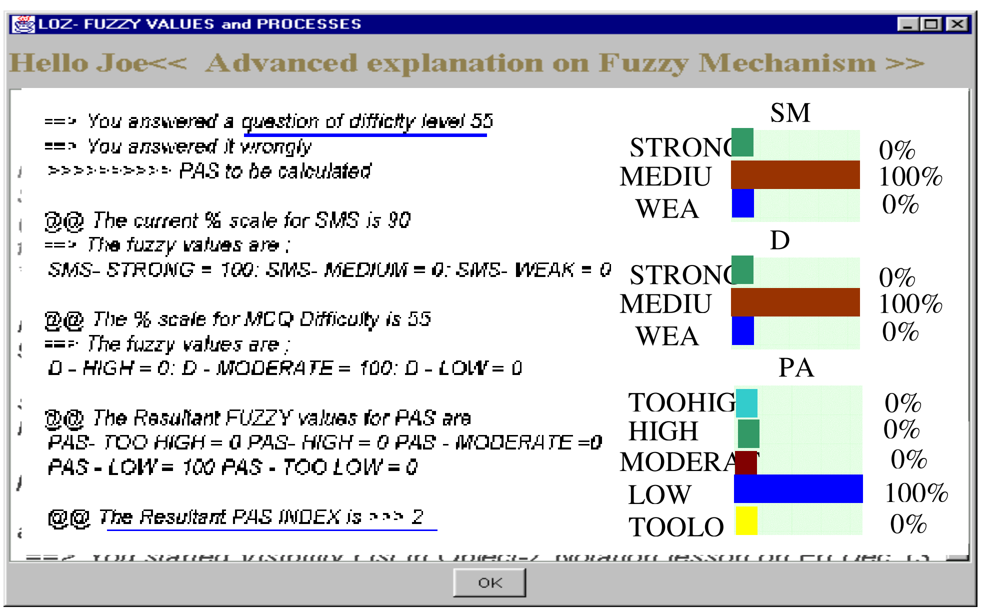
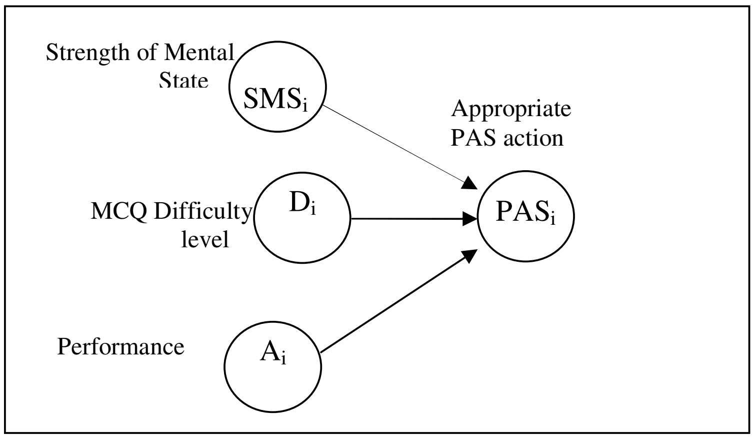


Figure 5(a). View the Learner Model's Actions: **5(b)** View the Learner Model's Mechanisms

The learner may inspect their learning process and learner model's decisions (Figure 5a). The explanation is given to the learner using every-day terms (for example, <Very Strong> instead of <90% Strong>). As the learner model is based on fuzzy logic, the fuzzy rule application process could be seamlessly described in the natural language. Moreover, some learners may want to take ultimate authority over their learning process, and may wish to investigate the whole fuzzy mechanism (Figure 5b). Note that figure 5b is meant only for who explicitly ask for technical details. We hope, encouraging this for an enthusiastic learner (it will be relatively easy to understand the fuzzy mechanism for a learner compared to other mechanisms such as Bayesian rules) could enhance their self-efficacy significantly. The hyperlink in figure 5b (under PAS index) will lead learners to a further level of detail through the fuzzy membership functions and fuzzy rule tables (figure 3a and 3b) and rule application processes.

(a)



(b)

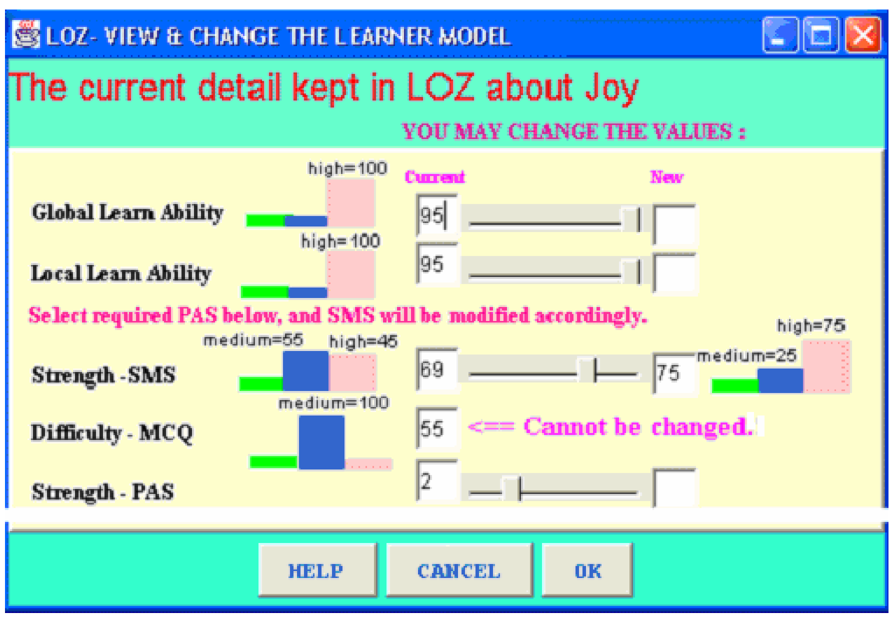


Figure 6(a). Causal Relations for PAS **6(b).** View & Alter Ability Measurements

The learner who might not be an expert and actually required detailed feed back for their failures (they might expect level 5 PAS action, but instead, they were given level 2 PAS action), might inspect the system and find (figure 5a) that the system ranked them as “Very Strong” for some reason (one reason might be the learner initially over-estimated his ability in pre-requisite lessons). Later, the learner, equipped with the information that the system has ranked them wrongly, may wish to alter the learner model. Figure 6a gives the belief network for the variable PAS. For a particular MCQ, after an answer is given, the difficulty level and performance are constant. Therefore, altering the PAS level will only affect SMS. An experienced learner might easily verify that for a given MCQ level, the PAS level decreases as SMS increases from weak to strong. Therefore, they know what to change and then what to anticipate (the possible impacts of their intended changes on the subsequent behavior of the system). For the beginners, however, judging their required level of feedback PAS (effect-variable) would be easier than estimating their own strength of mental state SMS (cause-variable). Therefore, the interface in figure 6b (which only cover wrong answers) allows learners to alter the SMS directly or, to set the desired PAS level (the SMS rate will be automatically adjusted). The values in the text boxes are automatically annotated by relevant bar charts.

It can be readily seen that usage of fuzzy model makes it easy to visualize the system's beliefs about a learner's different strengths. It is also straightforward to visualize the impacts on the system's belief due to any adjustments made by a learner. If learners feel later that they had rated themselves incorrectly, they may simply try fresh values. This trial-and-error feature mainly reduces the burden of learner model, but also allows the learner to have some control over the system's decisions.

Moreover, before or after learning a lesson, the learner can inspect the levels and types of scaffolding process, and can also modify this process for a lesson (some learners may perceive the scaffolding levels simply as different difficulty levels). For example, a lesson 'Visibility List' in LOZ has 3 scaffolding stages (figure 7). A gifted learner may change the starting stage to 3 and, therefore can avoid the easy early stages. Some lesson may have more than 5 stages. The incremental steps can also be altered. This facility will help learners to control their learning process, and in turn helps to develop their meta-cognitive abilities. If learners feel later that they had selected inappropriate start levels or stage steps, they may revisit the lesson and assign new values.

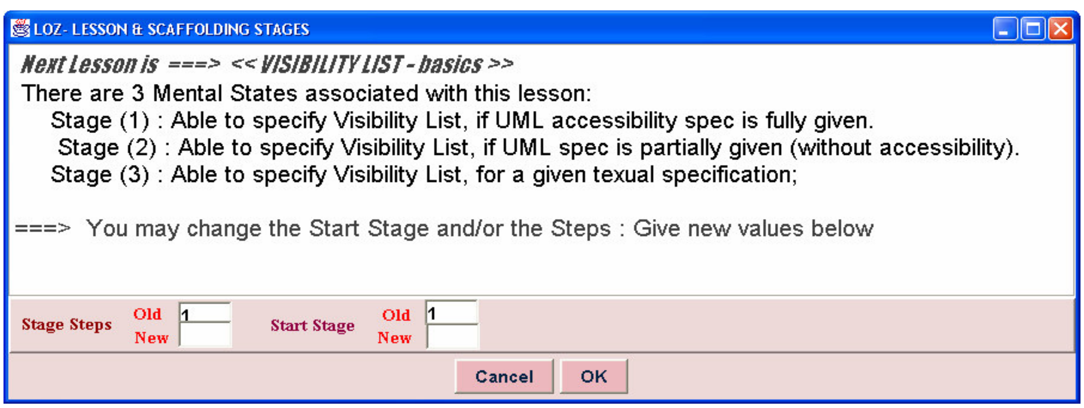


Figure 7. View & Alter Scaffolding Process

The system also keeps relevant information about the current and past learning activities of a learner, and if requested this information could be accessed by the learner. The top hyperlink in figure 5b will lead them to the recently attempted MCQ. They may try the same MCQ again or may process the feedback again. They can also go through the past scaffolding processes and critically analyze their own decisions in MCQs in different scaffolding levels.

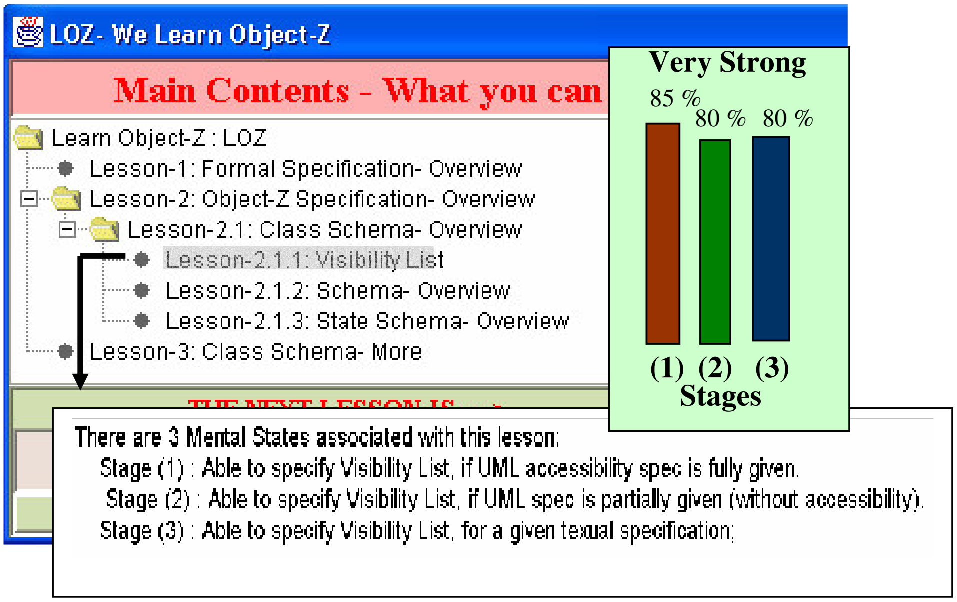


Figure 8. View & Compare the Performance Levels

Learners can view their past performance related to the concepts they have learned so far. The degree of strength of a mental state may be altered by a learner (however, the learner may need to convince the system- Bull [7] terms it 'negotiating'). If a group of students are involved, the learner may be able to compare their performance against the best, worst and average cases. This feature is also used by some other open systems for developing reflection habits in students [8].

5. Summary and Future work

Illustrating Fuzzy based learner model processes pertinent to typical learners is easy compared to explaining Bayesian networks. Providing graphical interfaces to alter numerical measurements and visualizing the impact immediately could encourage learners to use this facility. The users can alter the effect-variable straightly (rather than estimating a cause-variable in order to get the desired effect). Moreover, the system allows any changes to be reversed; and this facility encourages the learners to try and get confident in using those advanced features in trial-and-error fashion. Revealing the scaffolding process and encouraging the learners to change the intensity (number of stages) of this process is a novel idea. Moreover, we hope, revealing the whole fuzzy mechanism, as it is comparatively easy, to an enthusiastic learner would encourage their motivation significantly.

We have already developed a prototype (LOZ) that provides considerable facilities for exploring the learner model. Later, this facility will be extended to peers and instructors. Moreover, providing adaptive help on using these advanced features could be useful. For the purpose of academic learning, going beyond situated cognition, Laullilard states [16], *multiple contexts are necessary but are not sufficient; learners need to be engaged not just with their own experience, but with knowledge derived from someone else's experience*. By opening, we will provide facilities for learners to decontexualize the knowledge learned in multiple contexts and through social experience. Finally, a subjective evaluation should be carried out to estimate the effects of the different externalization processes mentioned in this paper.

6. References

[1] J. Self, "Bypassing the intractable problem of student modelling," presented at Intelligent Tutoring Systems conference, ITS'88, Montreal., 1988.

[2] S. Mohanarajah, R. Kemp, H., and E. Kemp, H., "Towards an Interactive Learning Environment for Object-Z," *Journal of Issues in Information Sciences and Information Technology Education ISSN: 1539-3585*, vol. Vol 1, pp. pgs 0993-1003, 2004.

[3] K. Hogan and M. E. Pressley, *Scaffolding Student Learning: Instructional Approaches and Issues*. The University of Albany, State University of New York: BROOKLINE, 1997.

[4] A. Collins, J. Brown, S., and S. Newman, E., "Cognitive Apprenticeship: Teaching the Crafts of Reading, Writing and Mathematics," in *Knowing, Learning and Instruction: Essays on honour of Robert Glaser*, L. Resnick, B., Ed. Hillsdale, NJ: Lawrence Erlbaum, 1990, pp. 453-494.

[5] S. Mohanarajah, R. Kemp, H., and E. Kemp, H., "Unfold the scaffold & Externalizing a Fuzzy Learner Model," presented at ED-MEDIA05 (accepted for publication), Montreal, Canada, 2005.

[6] A. Paiva, J. Self, and R. Hartley, "Externalising learner models" presented at World Conference on Artificial Intelligence in Education, Washington, DC, 1995.

[7] S. Bull, "Supporting Learning with Open Learner Models (Keynote Speech)," presented at Information and Communication Technologies in Education, Athens, 2004.

[8] J. Kay, "Stereotypes, student models and scrutability," *Intelligent Tutoring Systems, Proceedings*, vol. 1839, pp. 19-30, 2000.

[9] A. Mitrovic and B. Martin, "Evaluating the Effects of Open Student Models on Learning.," presented at Adaptive Hypermedia and Adaptive Web-Based Systems: Second International Conference, AH 2002, Malaga, Spain, 2002.

[10] J.-D. Zapata-Rivera and J. Greer, "Inspectable Bayesian student modelling servers in multi-agent tutoring systems," *J Int. J. Hum.-Comput. Stud.*, vol. 61, pp. 535-563, 2004.

[11] F. Pajares, "Overview of social cognitive theory and of self-efficacy.," Retrieved on 15/04/2005 from http://www.emory.edu/EDUCATION/mfp/eff.html., 2002.

[12] J. Martin and K. VanLehn, *OLAE: Progress Towards a Multi-activity, Bayesean Student Modeler.*: AACE, 1993.

[13] S. Mohanarajah, R. Kemp, H., and E. Kemp, H., "Intelligent Pedagogical Action Sequencing under Uncertainty," presented at AI-ED05 (accepted for publication), Amsterdam, Netherland, 2005.

[14] J. B. Mason and R. Bruning, "Providing Feedback in Computer-based Instruction: What the research tells Us. Retrieved on February 17, 2004 from http://dwb.unl.edu/Edit/MB/MasonBruning.html ." 1999.

[15] A. A. Hopgood, *Intelligent systems for engineers and scientists*, 2nd ed. Boca Raton, FL: CRC Press, 2000.

[16] D. Laurillard, *Rethinking University Teaching: A Conversational Framework for the Effective Use of Learning Technologies.* London:: Routledge, 2002.

SIMPRAC: Supporting reflective learning within a new computer-based virtual patient simulator.

Douglas CHESHER[1], Judy KAY[2], Nicholas JC KING[3]
1. Clinical Biochemistry, Pacific Laboratory Medicine Services, Royal North Shore Hospital, St Leonards NSW 2065. AUSTRALIA.
Ph: +61 2 9926 5524, Fax: + 61 2 9926 6395
Email: dougc@med.usyd.edu.au
2. Department of Information Technology, University of Sydney
3. Department of Pathology, University of Sydney

Abstract. Computer-based simulations aim to provide an authentic, interactive learning environment. However, there is evidence of case-specificity, and failure of knowledge to be transferred. We propose a promising means to address this limitation in the form of a reflective layer that is added to the simulation. This paper describes the SIMPRAC reflection model as applied to supporting learning of the management of chronic illness. One of the critical challenges for such a layer is user acceptance of the distraction with its break from the simulation activity. Accordingly, we have performed a study to determine user response to such a layer. We report an evaluation of SIMPRAC by 10 medical students, 5 general practitioners, and 2 specialists. This indicates that the review elements within SIMPRAC were well received and that learners see the value of SIMPRAC's reflective elements. We briefly discuss the relevance of this finding to the broader issue of the design of simulation-based learning environments.

Keywords. Medical Education, Computer Simulation, Reflection.

1. Introduction

Medical education is a natural place for simulation-based learning environments. Traditional medical training involves a long period of formal education followed by, or in association with, an apprenticeship, involving practice on human beings (1). This is hampered by the limited time that physicians have for teaching as well as the limited availability of patients as an educational resource (1, 2). After graduation, medical education typically involves a variety of forms such as rounds, educational meetings, conferences, refresher courses, and symposia. Unfortunately, these modalities are frequently ineffective in improving patient care through changes in physician behaviour (3). By contrast, interactive medical education sessions such as provided by simulation-based learning environments, involving practice of skills, can effect behavioural change (3). Such simulations can reduce the burden on teaching physicians and on patients, as well as providing opportunities for learning about aspects for which it is not feasible to give students direct experience (1, 2). A variety of simulation environments have been developed with varying levels of fidelity (2, 4-7). Nevertheless, it has been observed that many learners fail to translate experience from a specific simulated case towards broader expertise (7-10).

A promising possibility for tackling the problems of failure to transfer is to exploit the body of evidence that reflection and reflective practice have the potential to improve the effectiveness of learning from experience (11-13). Nevertheless, its use within medical education has been limited (11). Boyd and Fales state that the "process of reflection is the core difference between whether a person repeats the same experience several times, becoming highly proficient at one behaviour or learns from the experience in such a way that he or she is cognitively or affectively changed" (14). That is, by using the reflective process, learners have a greater chance of being able to generalize their knowledge so that it can be applied to new situations and experiences. Simulations differ from conventional and work-life learning opportunities in that the very artificiality of the learning environment opens the opportunity for forms of reflection that would not be available normally since the computer can track the learner's actions and can build up collections of data for cohorts of learning communities.

In light of these observations, we have established the SIMPRAC reflective model, which we describe in Section 2. We have implemented this in a web-based patient simulation. In Section 3, we briefly describe the implementation of this reflective layer as part of a simulation-based learning environment and the design of our evaluation of this reflective layer and Section 4 reports the results. In Section 5, we discuss the results and their importance for other simulation-based learning environments.

2. SIMPRAC reflective model

At one level, SIMPRAC is a simulation environment. However, it has been carefully designed to provide a layer of support for reflection based upon the model shown in Figure 2. Importantly, the approach we have taken in SIMPRAC is to first establish a base model of the desired interaction and then defined a corresponding reflection layer. It is these which

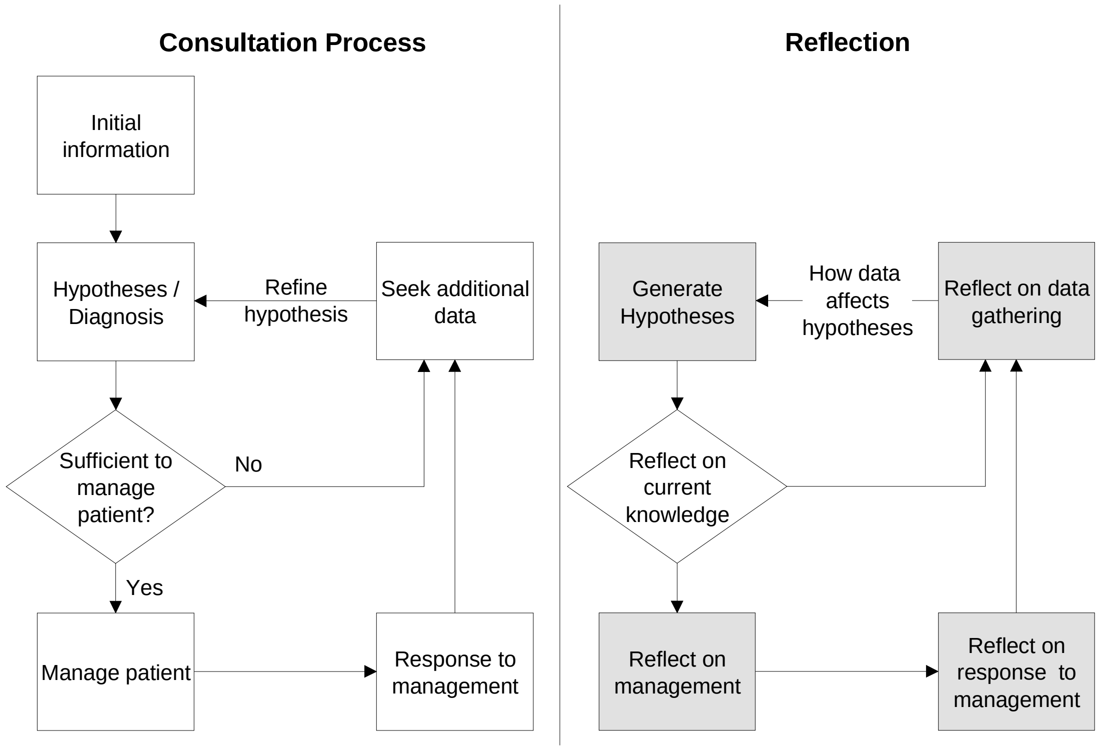


Figure 1. Model of the consultation process. Reflective processes explicitly supported in SIMPRAC

distinguish SIMPRAC from other simulations.

Our implementation of SIMPRAC as a simulation-based learning environment offers several points for reflection. However, the two major elements we describe here are the consultation reflection, and the comparative performance review. What we term the *consultation reflection*, is a novel reflection activity for learners to do at the end of each consultation.

Learners must revisit their actions and classifying each as one of, critical, relevant, or not relevant to the diagnosis, management and outcome of the patient, in light of their current knowledge. The second novel activity is the c*omparative performance review*, where learners view their own answers in the consultation reflection against those of an expert, the case author and the average for their peer group.

3. Implementation of SIMPRAC model and evaluation of a reflective layer

A SIMPRAC simulation begins with a short vignette, after which, the learner is able to take a medical history, perform a physical examination, request investigations and review results, as well as choose various management options, similar to other patient simulators, for example, DxR (5). However, in SIMPRAC, users are able to interact with the virtual patient over a predetermined number of consultations, and the patient's state in subsequent consultations is influenced by the management options chosen by the user. For a detailed description of user interaction with SIMPRAC, see (15).

To evaluate the reflective model of Figure 2, SIMPRAC was implemented to support a multi-consultation simulation environment intended to help medical students and staff develop their skills in managing chronic illnesses. The system supports reflection through:

- a multi-consultation architecture that enables the learner to see the effects of their actions from earlier consultations, this reflective element being typical of simulation environments where earlier actions affect later ones;
- the requirement that learners state their diagnostic hypotheses before examining the patient, requesting investigations, or choosing management options;
- support for notes;
- end of *consultation reflection*;
- end of consultation c*omparative performance review*.

The evaluation was conducted using a case that was designed to be unusual, unlikely to be familiar to most students or general practitioners. The case involved 4 consultations and a total 17 possible states. This was evaluated, using a simplified think-aloud technique (16) with ten medical students, five general practitioners, and two expert consultants. The consultants were both chemical pathologists with experience in the management of lipid disorders.

After each participant had completed their period of using SIMPRAC, they answered a three part questionnaire. Part A included basic demographic information. Part B was a series of semantic differential statements relating to each aspect of the interface, and about the reflection. Each statement was scored using a 5 point Likert scale varying from, "strongly disagree" to "strongly agree". Part C included open ended questions.

Analyses were based on a two tailed Students t-test, assuming unequal variances. Linear associations between properties were assessed with Pearson's correlation coefficient. Associations were considered significant for p less than 0.05. Non-linear associations were assessed for selected data sets, by first curve-fitting the data, then recalculating the correlation coefficients on the predicted values.

4. Results

One important aspect of reflection is whether learners are prepared to take the time out from the main activity and reflect on their recent activity. Clearly, if the SIMPRAC approach is to be effective, learners must be willing to engage in the reflective activities. Our study of users interacting with SIMPRAC involved careful observation of the way that learners responded to the reflective activities and their attitude to them. In the scope of this paper, we restrict the results to just the two novel forms of reflection support: the *consultation reflection* and *comparative performance review*. Essentially, the first calls upon the learner to stop the main activity at a logical stopping point and to look back at their actions, assessing these and then,

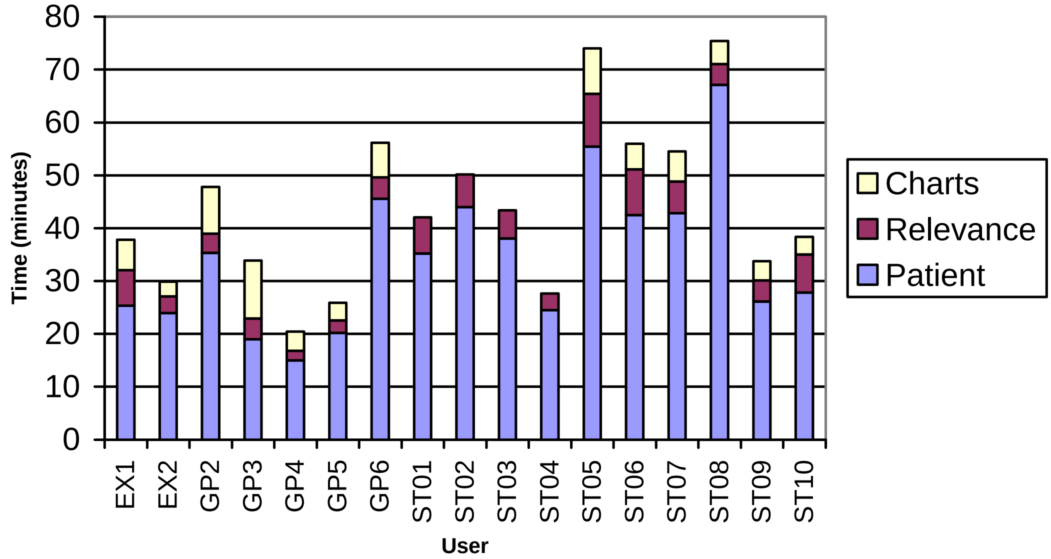


Figure 3. Total time each user spent interacting with the patient, reviewing the relevance of their actions (reflection), and reviewing the charts in the first consultation. Data on time in chart review for ST01, ST02, ST03, and ST04 could not be included due to inadequate data logging during the first

the second involves examining how the learner's own actions compared with those of the author of the simulation as well as against their peer group's performance. Although our SIMPRAC evaluation is in the area of medical management, these aspects of the reflective layer are likely to be generalisable to many simulation-based environments. An important concern associated with the addition of a reflective component is that it affects the flow of the simulation experience: it stops the learner from engaging in the case and asks them to step back. We were concerned that this would be resented by learners. We now report the relative time that our study participants spent in reflective activities, their attitudes to those activities and the relationship between the time spent reflecting and reported attitudes.

Three students (ST01, ST02, and ST03) completed two consultations. One student (ST10), and two general practitioners (GP3, and GP4) completed three consultations. All other users completed four consultations. Since patient outcomes depended on the management actions taken, and these differed for different users, only the first consultations is strictly comparable across users.

Figure 3 shows the time each user spent interacting with the simulated patient as well as in each of the two reflective activities. This data is summarised in Table 1. General practitioner (GP) 3 spent relatively more time reviewing the charts at the end of the first consultation. However, the absolute amount of time spent undertaking this activity was not much greater than that of other users. ST08 was very interesting. Although she spent little time involved in the reflective review, during the think-aloud sessions, she was noted to continually vocalize her thoughts and hypotheses to a much greater degree than the other users. As a consequence, this user took much longer than the others to complete the case.

Table 1. Summary of time spent by each group in each activity (Mean ± SD)

	Total Time [1]	Patient Interaction	Consultation Reflection	Comparative Performance Review
Medical Student	123 ± 34	40.4 ± 13.3 [2]	6.1 ± 2.2 [3]	5.1 ± 2.0
General Practitioner	102 ± 29	27.0 ± 13.0	3.2 ± 1.0	6.7 ± 3.2
Experts	86 ± 17	24.7 ± 1.0	4.9 ± 2.5	4.3 ± 2.1

[1] Time from the start of the first consultation to the start of the second consultation. Due to the small numbers and large variances, the differences were not statistically significant.
[2] The students spent significantly more time interacting with the patient than the experts (p=0.005)
[3] The students spent significantly more time than the general practitioners in *consultation reflection* (p=0.003).

In the consultation reflection, most users found that having to classify their actions as critical, relevant or not relevant helped them to reflect on their activity (Figure 4). Only two users gave a negative response to this question. There was no obvious linear relationship between the time users spent reviewing their activity, and the degree to which they stated the activity supported reflection (r = -0.185, p = 0.477). As illustrated in Table 1, the general practitioners

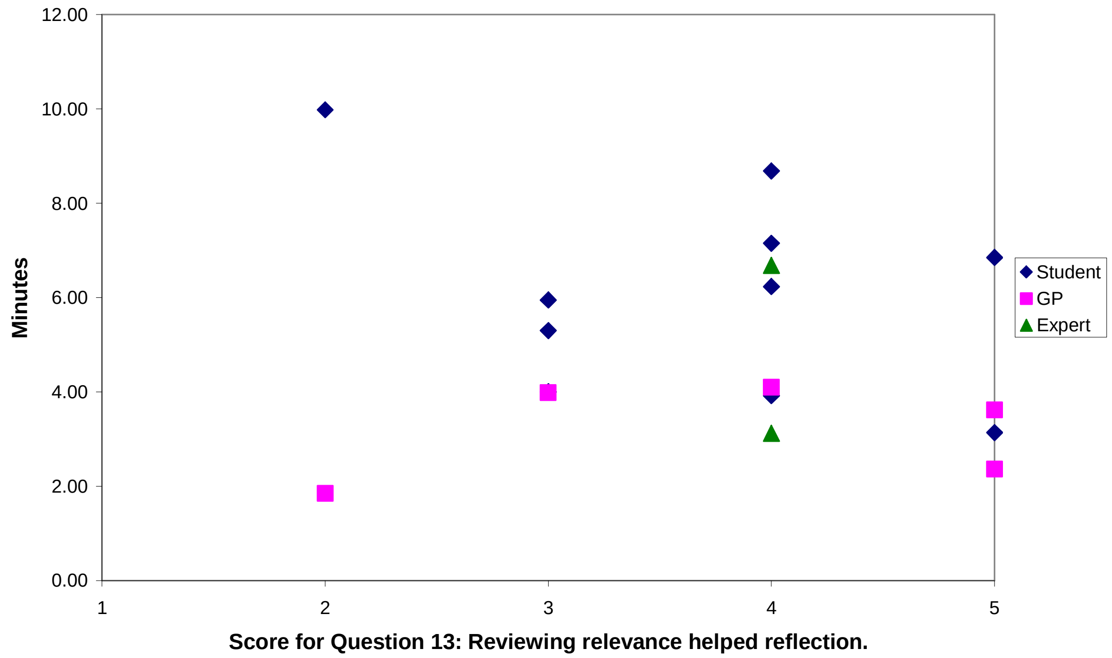


Figure 4. Relationship between the duration over which the user reviewed the relevance of their actions in consultation 1 and the score for question 13 of the questionnaire. Question 13 was, "I found having to classify the importance of my questions and actions helped me to reflect on their usefulness."

did spend less time in consultation reflection than the medical students (p=0.003). Furthermore and very importantly, there was no clear relationship between the time users spent reviewing their activity and their stated level of frustration (r = 0.151, p = 0.562). Most participants reported the consultation reflection and comparative performance review helped them to reflect on the important diagnostic and management issues. Again, there was no correlation between the extent to which they found the consultation reflection screens helpful and the duration over which they spent reviewing their activity (Figure 5) (r = 0.172, p =

0.508). While there were some individuals who clearly found the consultation reflection process frustrating, this was not related to the total number of actions chosen by users in the first consultation (r = 0.154, p = 0.556). The total number of actions includes the total number of questions asked, examinations performed, investigations ordered and management options selected in a single consultation. However, those individuals who found the consultation reflection activity most helpful, also found it least frustrating (r = -0.609, p = 0.009). Together, these data illustrate that it is still possible to find the learning process helpful, even though it may be frustrating.

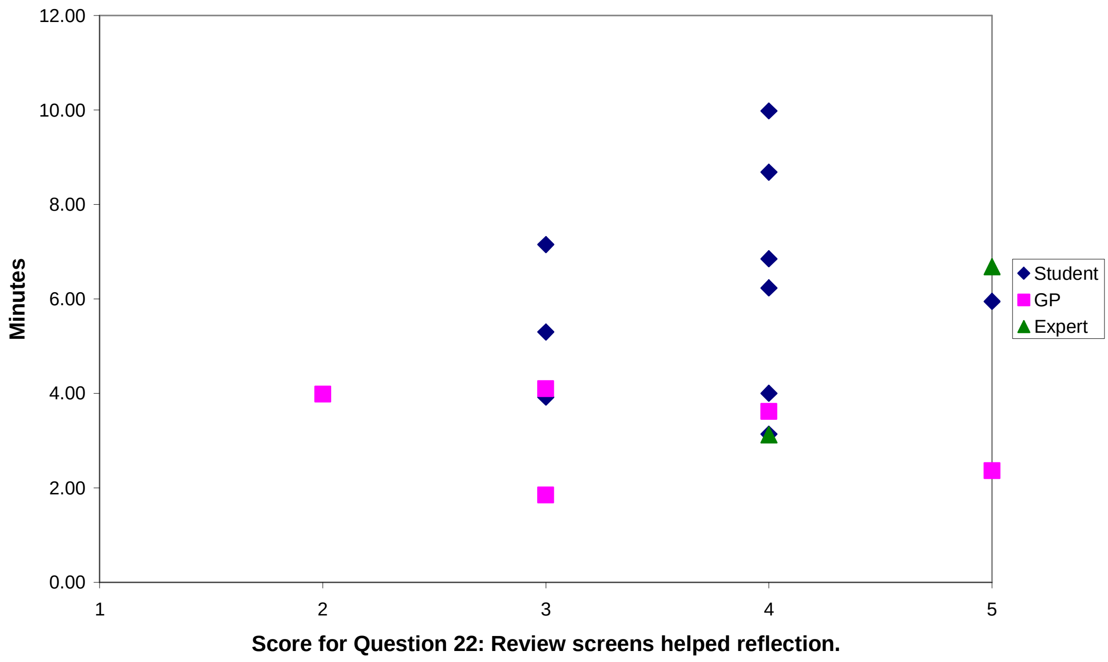


Figure 5. Relationship between the duration over which the user reviewed the relevance of their actions in consultation 1 and the score for question 22 of the questionnaire. Question 22 was, "The review screens helped me reflect on the important diagnostic and management issues involved in this case."

From the questionnaire data, with the exception of ST03, GP04 and EX2, users were able to use and interpret the bar chart that showed the overall comparisons between the user and their cohort. However, four users (ST03, ST07, GP4, and EX1) experienced difficulty in determining which of their specific actions had been classified as critical, relevant, or not relevant, as well as which critical or relevant actions they had failed to perform. EX1 suggested this information should be more accessible, rather than requiring two mouse clicks.

While most users stated the comparative performance review screen helped them reflect on the important diagnostic and management issues, there was no correlation between the extent to which they held this opinion and the time spent in this activity (r = -0.291, p = 0.336), or the number of different charts they reviewed (r = 0.195, p = 0.522) at the end of consultation one.

Figure 6 compares the time each user spent in consultation reflection with the time they spent in comparative performance review, i.e., reviewing the charts and associated information at the end of consultation one. Overall, there was no clear association between these times. However, this data suggests that there is a separate linear relationship for the students (r = 0.666, p = 0.148) and the general practitioners (r = 0.827, p = 0.084), although, due to the small numbers in each of these subgroups, this did not reach statistical significance. As there was incomplete log data for the first four students, so only six students are represented in Figure 6. Figure 6 clearly indicates that the students and experts spent relatively more time reviewing the relevance of their actions than the general practitioners. In contrast, the time spent reviewing the charts was similar for both the students (5.1 minutes, n = 6) and the general practitioners (6.7 minutes, n = 5) (p = 0.37). This relationship is clearly reflected in the difference in the slope of the two regression lines for the students and general practitioners, respectively.

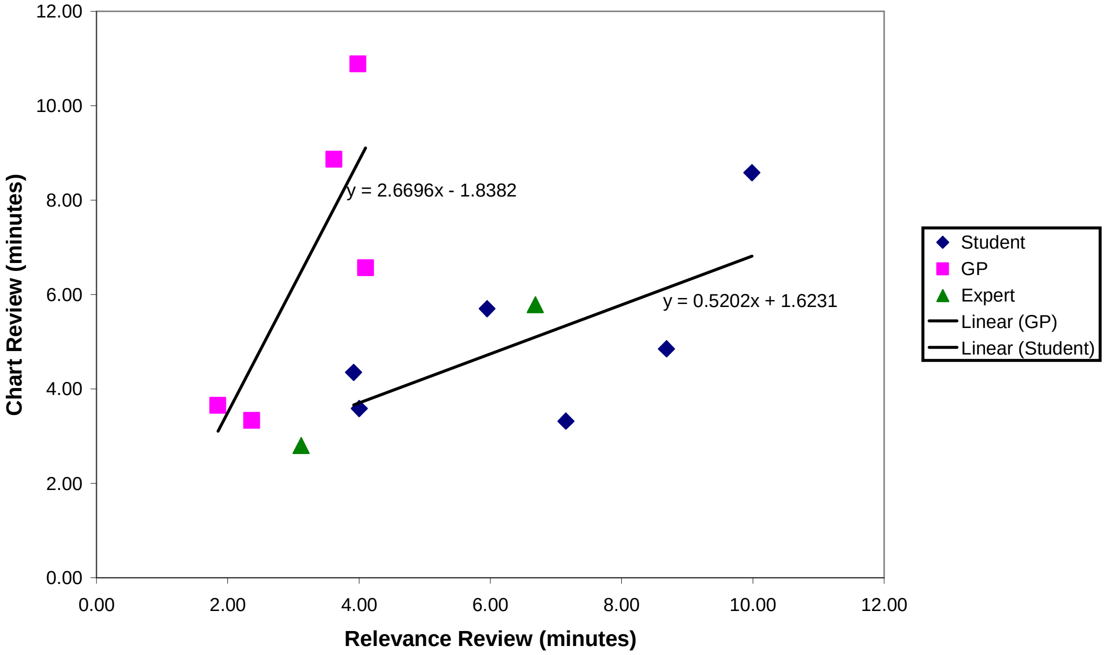


Figure 6. Time each user spent reviewing the relevance of their actions versus the time they spent reviewing the charts.

Excluding the data for the four students where there was incomplete log data, all users spent much more time interacting with the patient and reviewing their activity in the first consultation (44.9 ± 17.4 minutes, n=13) than in the second (11.1 ± 4.1 minutes, n=13), third (8.4 ± 2.9 minutes, n=13) or fourth (6.7 ± 3.6 minutes, n=10) consultations. This was an expected finding given that most of the diagnostic activity occurred in the first consultation.

5. Discussion and conclusions

The evaluation indicates that the participants were willing to devote time to the reflection activities and they considered that the reflective activities helped them to reflect on their experience with the simulation. At the same time, there were some interesting user behaviours. In particular, Student 8 (ST08) spent less time than most users in both reflection activities. Yet this student spent the longest time with SIMPRAC, constantly rephrasing the patient's issues. In the schema of Schön (13), this student performed reflection-in-action, reflecting on an activity as it unfolds. While reflection-in-action is required for good practice, learning is best where firstly, one reflects on the actions that have been taken, such that more appropriate action can be taken should a similar circumstance arise (reflection-on-action), and secondly, where the learner thinks about what they were reflecting on at the time of the activity (reflection on reflection) (12, 13).

SIMPRAC is founded on the considerable literature indicating the value and importance of reflection for improved learning (13, 14, 17, 18). Our work represents an important start on the improvement of learning outcomes from simulation environments. Essentially, while simulation environments are engaging and can be quite authentic, this is not enough for deep learning (7). The level of engagement may discourage learners from taking time to reflect enough to learn effectively. The two reflective elements are quite simple and general enough to be applied in most simulation-based environments.

At this stage, we can conclude that the reflective elements appear to be sufficiently engaging and meaningful for users to take time to stand back and reflect on their actions and to self-assess their performance. SIMPRAC enables learners to do reflection both in terms of the precise details of their consultative actions, right down to the point of each individual action. It also supports cohort comparisons. These reflective elements, both the *consultation reflection* on actions and the *comparative performance review*, could well be used as a foundation for design of a reflective layer for other simulations. With the consultation reflection on actions, the designer of the simulation needs only to identify those actions which the learner should reflect upon, and then the author of the simulation needs to code a set of standard answers for comparison. The comparative performance review is potentially more difficult, as it requires data from the relevant groups of learners. Even this should be a modest additional cost for builders of simulation-based learning environments. Of course, in the long term, it will be important to run careful studies which compare the deep, long-term and transferable learning that is achieved both with and without reflective elements attached to a simulation.

In summary, SIMPRAC represents an exploration of two forms of support for reflection, both widely applicable and of modest cost, which have the potential to enhance simulation-based learning environments with a reflection layer.

6. References

[1] Issenberg SB, McGaghie WC, Hart IR, Mayer JW, Felner JM, Petrusa ER, et al. Simulation technology for health care professional skills training and assessment. JAMA 1999;282(9):861-866.

[2] Bergin RA, Fors UGH. Interactive simulated patient-an advanced tool for student activated learning in medicine and healthcare. Computers & Education 2003;40:361-376.

[3] Davis D, O'Brien MAT, Freemantle N, Wolf FM, Mazmanian P, Taylor-Vaisy A. Impact of formal continuing medical education: Do conferences, workshops, rounds and other traditional continuing education activities change physician behaviour or health care outcomes? Journal of the American Medical Association 1999;282(9):867-874.

[4] Hayes KA, Lehmann CU. The interactive patient: a multimedia interactive educational tool on the world wide web. MD Computing 1996;13(4):330-334.

[5] Myers JH, Dorsey K, Benz E. DxR Clinician. Carbondale: DXR Development Group; 2001.

[6] Bloemendaal PM, Egggermont S, Schoonderwaldt EM, Van Baalen JM. Medical training with the dynamic patient simulator. In: Slice of Life; 2002; Toronto; 2002.

[7] Bond WF, Deitrick LM, Arnold DC, Kostenbader M, Barr GC, Kimmel SR, et al. Using simulation to instruct emergency medicine residents in cognitive forcing strategies. Academic Medicine 2004;79(5):438-446.

[8] Elstein AS, Shulman LS, Sprafka SA. Medical problem solving: A ten year retrospective. Evaluation & the Health Professions 1990;13(1):5-36.

[9] Fitzgerald JT, Wolf FM, Davis WK, Barclay ML, Bozynski ME, Chamberlain KR, et al. A preliminary study of the impact of case specificity on computer based assessment of medical student clinical performance. Evaluation & the Health Professions 1994;17(3):307-321.

[10] Eva KW. On the generality of specificity. Medical Education 2003;37:587-588.

[11] Branch WT, Paranjape A. Feedback and reflection: teaching methods for clinical settings. Academic Medicine 2002;77(12 Pt 1):1185-8.

[12] Ruth-Sahd LA. Reflective practice: A critical analysis of data-based studies and implications for nursing education. Journal of Nursing Education 2003;42(11):488-497.

[13] Schön DA. Educating the Reflective Practitioner. Toward a New Design for Teaching and Learning in the Professions. San Francisco: Jossey-Bass; 1987.

[14] Boyd EM, Fales AW. Reflective learning: key to learning from experience. Journal of Humanistic Psychology 1983;23(2):99-117.

[15] Chesher D, Kay J, King N. A web-based virtual patient to support reflective learning. In: Aleven V, Hoppe U, Kay J, Mizoguchi R, Pain H, Verdejo F, et al., editors. 11th International Conference on Artificial Intelligence in Education - Supplementary Proceedings; 2003; Sydney: School of Information Techologies, University of Sydney; 2003. p. 209-218.
[16] Nielsen J. Usability Engineering. Boston: AP Publishing; 1993.
[17] Sobral DT. An appraisal of medical students' reflection-in-learning. Medical Education 2000;34(3):182-187.
[18] Boud D, Keogh R, Walker D. Reflection: Turning Experience into Learning. London: Kogan Page; 1985.

A Framework for Designing and Analysing Open Learner Modelling

Susan BULL[1] and Judy KAY[2]

[1] *Electronic, Electrical and Computer Engineering, University of Birmingham, U.K.*
[2] *School of Information Technologies, University of Sydney, Australia.*
s.bull@bham.ac.uk, judy@it.usyd.edu.au

Abstract. Recently learner models have been opened to the learners they represent. However, as yet there is no standard way of describing and analysing open learner models. This is in part due to the variety of issues that can be important or relevant in any particular model. Nevertheless, this lack of a framework to discuss open learner models makes it difficult to compare the features of open learner models in different systems. We believe this is a serious barrier to the effective use of open learner modelling. This paper presents such a framework and gives an example of its use to describe a system.

1. Introduction

Self [1] points out that various components of learning environments can be usefully made open, stating that "it is only the fact that learner models have traditionally been 'closed' that opening them up is such a big deal". However, this does not diminish the importance of open learner models.

Open learner models are models of the user that are available for viewing - usually by the learner, and sometimes also by others. The externalisation of the learner model can be in a simple form such as a skill meter showing learner progress as a subset of expert knowledge [2]; or the information presented can be more complex, such as a graphical externalisation of a Bayesian network [3]; a hierarchical tree structure [4]; a conceptual graph [5]; or textual descriptions of knowledge and misconceptions [6]. Students can even be involved in the maintenance of their learner model, for example by being able to edit it [4], or by negotiating the contents of the learner model with the system [5,6]. Furthermore, learner models can be opened not only to the learner modelled, but also to other users (e.g. peers or instructors) [7].

As indicated by the above, interactions with an open learner model, and presentation formats of open learner models, differ. Other differences include the extent to which a learner model is accessible to the learner; who may initiate access (learner and/or system?); whether the learner has access to information regarding uncertainty in the model; flexibility of access to the model. In fact, the seemingly simple matter of making a learner model open turns out to involve a quite complex range of choices, each with significant implications. This has important implications for the designers of new personalised teaching systems with open learner models. It also makes it harder to see the important differences between ways that existing systems have made use of openness of learner models. Furthermore, descriptions of open learner models do not follow any standard (or even any pattern). This paper addresses the need for an open learner modelling framework for use by researchers, for the description and analysis of open learner models, and to support the design of open learner models in new systems.

2. The SMILI☺ Open Learner Modelling Framework

Table 1 shows our proposed SMILI☺ framework for *Student Models that Invite the Learner In*. The columns correspond to purposes or goals of the openness of the learner model: these are the reasons *why* a system would make the model open. The rows correspond to the elements and means of achieving openness. The cells marked with 'X' indicate particularly strong connections between purposes and elements. In the remainder of this section, we discuss several elements of the framework to give the flavour of each part.

The column labels indicate important *purposes for openness:* improving accuracy of the learner model by allowing learners to contribute information; promote reflection, an important metacognitive foundation for learning; help plan and/or monitor learning based upon the foundation of information available in the learner model; facilitate collaboration because partners can improve understanding of themselves and each other by gaining information from their respective learner model(s); afford learners greater control over learning through greater control over their learner model; and the privacy issue of the right to view data about oneself. A starting point in the design or analysis of openness of the learner modelling in a system is to treat this set of purposes as a checklist to review, carefully considering whether the system needs each (or, alternatively, carefully considering whether it can be argued that the system does *not* need each). Once the purposes, or goals, of openness are identified, one needs to determine how to achieve them. The rows identify issues to consider at this stage. They are numbered for reference. The 0-th is of a different character from the others, which all have choices indicated in the second column. We now briefly review each of the purposes, with discussion of the elements that are important, hence marked with a cross in the table. At this stage some of the discussion is necessarily speculative, and will need to be refined and extended as users of the proposed framework apply it to their systems.

Centrality of Openness of the Model:
Openness may be more or less central to a system's aims. Any of the listed purposes for opening the model could make the openness central. However, the learner's right to view their model is of a different character from the other purposes in that it would generally not affect the learner's normal interaction with the system.

Accuracy of the Model:
We discuss this in some detail to illustrate the interpretation of the figure and the framework. First, we consider the three aspects of element 1, the extent of openness. There is an especially strong connection between opening the model to increase its accuracy, and allowing complete access. If a learner is to contribute information to improve model accuracy, they need to be able to find out what is there. Particularly if the model is small or easy to understand, complete access may improve accuracy. However, partial access can also enhance accuracy, and partial access can therefore be helpful. Indeed, for complex learner models, the complete model may be so overwhelming that the learner could not easily correct problems. In this case, partial access may be more effective.

We now consider the second aspect of the extent (1), the types of modelled aspects: knowledge and difficulties. To improve model accuracy by allowing input from the learner, the learner should have access to representations of both their knowledge (content known

Table 1. The SMILI☺ Open Learner Modelling Framework

Open Learner Model	*Properties*	*Accu-racy*	*Reflec-tion*	*Plan / Monitor*	*Collab-oration*	*Control*	*Right to view*
0. Centrality of openness							
1. Extent of model accessible	Complete	X				X	X
	Partial	X					
	Knowledge	X	X	X		X	X
	Difficulties	X	X	X		X	X
	Learning issues	X				X	X
	Social issues	X				X	X
	Preferences	X				X	X
	Other	X				X	X
2. Presentation	Textual (i.e...)						
	Graphical (i.e...)						
	Summary						
	Overview						
	Targeted Details	X				X	X
	All Details	X				X	X
3. Similarity to underlying representation	Identical	X					X
	Similar	X					X
	Different						
4. Access to uncertainty	Complete	X				X	X
	Partial			X			
	None						
5. Role of time	Previous						X
	Current	X	X	X		X	X
	Future						X
6. Access method	Inspectable		X	X		X	X
	Co-operative	X					
	Editable	X				X	
	Negotiated	X				X	
7. Access initiative	System initiated	X					
	Learner initiated	X		X		X	X
8. Access to sources of input	Complete	X				X	X
	Partial	X					
	None						
	System	X	X	X	X	X	X
	Self	X				?	X
	Peer					?	?
	Teacher					?	?
	Other program					?	X
	Other					?	?
9. Control over accessibility (to others)	Complete				X	X	X
	Partial				X		
	None						
	System					X	X
	Self	X				X	X
	Peer					?	X
	Teacher					?	X
	Other program					?	X
	Other					?	X
10. Awareness of effect of model on personalisation	Complete					X	
	Partial			X	X		
	None						
11. Flexibility of access	Complete					X	
	Partial						
	None						

or knowledge ‘level’), and about problematic areas. This may range from an indication of the areas in which they are having difficulty, to specific misconceptions. Only if the learner has access to positive and negative data (if both are modelled), can they make an informed decision on the correctness of the data. Of course, if the model is an overlay, only ‘known’ concepts, or an estimate of the knowledge level, will be available. Increasing the accuracy of such a model will probably be focussed around knowledge only. (It is, however, possible that a learner could provide information that they do *not* know X, if X is not a precise detail that requires comprehension before articulation.) For accuracy, it is also important that the learner have access to all other model data.

The third aspect of extent shows four quite different types of modelling information. For similar reasons to the above, all these aspects of the learner should be available to improve the model accuracy. At this point, we note that it is quite feasible to have the model only partially open, but still to make those parts include all types.

We now consider the role of presentation (2) of the model for the goal of accuracy. The framework identifies only that the learner must be able to see at least targeted details, in the case of partial access, or the full model otherwise.

A model should be available in a form similar or identical to the underlying representation (3) for greater accuracy, as long as the presentation is understandable.

Complete access to uncertainty (4) is important for accuracy. For example, suppose a learner is aware they do not fully understand a particular concept. If they see an indication of the system's uncertainty of this aspect, they can make more sense of the model. By contrast, they may not accept that the model shows they do understand it (if the system withholds the information that it is uncertain).

The role of time (5) for accuracy is shown with current data being important.

The learner must have access (6) to the model to improve accuracy. They must be able to provide information to the model, or about the model, otherwise they cannot influence its accuracy. Usually this involves model inspection (not necessary where learner and system co-operate in the contribution of data, but the learner cannot view its contents). Co-operative models and negotiated models (where student and system discuss and agree the contents) are usually designed at least in part to improve accuracy. Similarly, this is usually an intended purpose of editable models (which allow the student to change the model directly), though here control is given to the learner.

Mixed initiative (7) is important for accuracy. The system should be able to request information when required. It is also important that the learner can choose to offer information; openness motivated by the goal of accuracy, is based on the expectation that the model may be wrong, and that the student may be able to help.

Similarly, the learner should have some degree of access to different sources (8) of input. If the model is constructed mainly by the system or jointly by system and student, the situation is straightforward: the learner needs access to both sources. However, if the model also contains data from others, e.g. teacher, this assessment may be more accurate. A similar argument could be made for peer contributions. Conversely, the student's self-assessment may be correct. However, as it is difficult to resolve discrepancies between peer- and self-assessment (and indeed, a learner may change a peer-assessment inappropriately), for the purpose of an accurate model, access to peer contributions may not be essential. Unless the system can accurately determine the reliability of evidence from different sources, the only point about which it can be confident is that the student's self-evaluations probably reflect their beliefs.

Control over who (or what) else can access the information (9) is largely not helpful for accuracy. Indeed, learner control over accessibility to others could result in decreased accuracy of the model. Certainly if the learner could withhold some data from the system, this would limit its ability to accurately model them.

Awareness of the effect of the model on adaptivity (10) is in itself not important for accuracy. However, such awareness may raise motivation to reach an exact model.

Finally, the flexibility of access (11) - i.e. are there choices about how to access the model? - is not important for accuracy, though it might be desirable that a learner can access precisely those (and only those) contents required, presented in the manner they wish to view them, and perhaps interacting using a method they prefer.

This more detailed discussion of the elements that are important for accuracy of the model was intended to introduce the elements that characterise openness and support the various motivations for opening learner models. For the rest of this section, we will focus on the less obvious parts of the framework.

Reflection:
Where the aim of opening the model is to promote reflection, access to information about knowledge and difficulties (1) is important. A learner could reflect simply on information about their knowledge, as this could influence their confidence in their progress, while perhaps also helping to raise awareness of what they do not know. However, if the system models difficulties or misconceptions, reflection on these is important. Whether this information should be complete or partial will depend on the particular system. Similarly, in some systems it may be helpful to encourage reflection on other model contents, though this may not apply as a general rule.

It may seem surprising that we do not show that reflection, in general, requires a particular level of access extent (2) or that the model presentation to be similar or identical to the underlying representation (3), or that it is different. The critical factor is that the model is easily understood.

It may not be necessary to have access to uncertainty (4). If the learner notices disparities, this may lead them to reflect further! However, a question of trust in the utility of the model as a source of information for reflection may arise if they perceive it to be withholding information.

The role of time (5) for reflection on knowledge must focus on the current model, as this gives the learner the clearest indication of where they are. In some systems, reference forwards and backwards in time might be helpful.

To reflect on their learner model, the learner must be able to see the contents (6) – i.e. an inspectable model is assumed. This may be sufficient to encourage reflection. Other access methods are not critical.

While it is desirable that a system prompt (7) the learner to reflect on their model where this would be useful, and that the learner can undertake unsolicited viewing of the model, this combination is not essential. Conceivably, either approach separately could work well in a learning environment designed to promote learner reflection.

Of greatest importance, in most systems, students will need access to the system's inferences about their beliefs (8).

The remaining elements are not shown as important for reflection. In essence, the reflection column of the table indicates that the critical aspects for supporting reflection are that the learner be able inspect (6) the system's view (8) of their knowledge and difficulties (1) at the current time (5).

Planning/Monitoring Learning:
This goal is very similar to reflection, in that it involves the learner in assessing how they are doing, and how to use this information to decide on future learning goals. Accordingly, this discussion deals with the differences between this column and that for reflection. One difference is the greater importance of at least partial access to uncertainty (4). This supports decisions about the reliability of information in the model, which is used to help plan and monitor progress. It is likely that partial access will be sufficient, as it is only knowledge of the existence of uncertainty, or an indication of the extent, that the learner really requires for this purpose.

To monitor or plan learning, learners need to access the current model. This may be inspectable only. Users must be able to initiate inspection (7) to plan and monitor effectively, as autonomous learners. Most important is access to system inferences, as the system will have been designed to facilitate learning in an effective manner.

It is important the learner is aware of the effect of their model on the interaction (10), otherwise they will not be able to use their model data effectively in planning.

Collaboration:

Although they could be potentially relevant in a specific system, most open learner modelling properties are not essential to support collaboration - e.g. it is not crucial that users have complete or partial model access (1), as long as they have sufficient access according to the purpose of the open learner model.

Although access to representations of knowledge (1) is likely to be important in many environments, to support collaboration a system may instead be designed to match learners, or support them in some other way, according to social issues or preferences. Thus there is no pre-specified link with the extent of the model accessible.

Similarly, presentation method (2) is not strongly linked to purpose of collaboration (though it may be important in specific settings). A number of problems can arise, e.g. if a group model is presented to co-present peers, what format should the presentation take? Such issues may be resolved in different ways in different systems. Furthermore, in a collaborative context, users may still view their model individually. As in the cases described previously, it is difficult to prescribe a presentation format that would generally suit all students, all domains and all likely educational aims.

To support collaboration, it is not necessary that the presentation is identical or similar to the underlying representation (3). What is important is that the student can understand the model for the purpose that it was made accessible in the environment.

Access to uncertainty about representations (4), although potentially helpful in some settings, is probably not crucial in a collaborative environment.

With the other listed purposes of an open learner model, access to the current model (5) is important. In a collaborative environment this may also be so, but it may be, for example, that the learner can view the predicted future state of a peer, and then choose to work with that student once they have reached that state. Alternatively, a student might note that someone had previously had difficulties similar to their own, that are no longer problems according to their current model. This could indicate to the learner that this person may be particularly suitable to help them.

As learners may not only have access to their own model, issues such as whether an individual can edit another person's model (6) need to be considered; and whether all peers have the same kind of access to the same learner model.

Whether learner or system initiates model access (7) will depend on the system's aims. For example, in a system that matches partners, if it is a student's responsibility to find a partner by comparing learner models, they will need to access them. However, if the system does the matching, it may show learners the relevant models to help collaboration, once they are matched and maybe already working on a problem.

In a collaborative environment there are potentially many sources of data for the leaner model (8), as peers could be involved in providing data about each other. A learner model may be an individual model and/or it may be a model of all group members combined in some way from their individual models, or maintained according to group decisions. The extent of access to sources of input to the (individual or group) model is less critical in general, but as for other uses of open learner modelling, information modelled by the system is central to an open learner model to support collaboration. If peers or others contribute data to an individual's model, this may be accessed if it helps the collaborative experience, but it is not usually essential that learners know that specific data comes from specific individuals (or peers in general).

It could be argued that the learner should have control over accessibility (9) of their model to peers (or others). This may be complete or partial - e.g. a learner might give permission for their data to be available to any peer, or they might restrict their individual model from (some) others, while it is still available as part of an aggregate model comprising data from all students. This, however, may be more to do with privacy than facilitating collaboration. From the perspective of *collaboration for learning*, at least in some systems, access to the models of others might be useful.

If the model is to be used to help students collaborate, to understand its relevance in a collaborative environment, students should have some awareness of its effect on personalisation (10). This is unlikely to be complete awareness, but it should be sufficient for students to understand the model's purpose. This should help motivate interaction with others if some contents available relate in some way to other learners.

Control Over the Model:

Control may refer to a learner's control over learning (related to learner autonomy); the interaction; or the model. These notions are linked - if the purpose of the model is understood, control over the model will lead to some degree of control over the interaction, resulting in increased control over learning.

Where learner control is central to a system (0), many properties of open learner modelling are key. If the aim is to allow users greater or full control over the model, interaction and/or learning, they should have complete access (1) to all contents.

Presentation mode (2), as other cases discussed, must be understandable with reference to controlling learning, the interaction, or the model. However, the learner should be able to access all details, or just those relevant for a particular purpose. The latter holds true especially if the model is large or complex - presentation of too much information may not be helpful, and may even be confusing. In practice it could result in little learner control being effected through the open learner model. A summary of contents or an overview of the model is not necessary (though it may be helpful).

Similarity of the presentation to the underlying representations (3) is not relevant, for similar reasons to those discussed concerning the potential complexity of the model being overwhelming if rendered at the wrong level or form.

It is important that learners can retrieve information about the uncertainty (4) of model data, since they can only have full control if they have all relevant information.

It is most important that the learner can access their current model (5), as it is their current learning or interaction over which they can have immediate control. Previous models, as a representation of previous states, will often not be manipulable. This may not apply, however, if the system designer accepts that a student's control over their model should be ultimate, or that the student may be able to correct something that was inaccurately represented in a previous model. Nevertheless, a question is whether a historical record should be alterable later. Arguments for access to future models are similar, with the additional difficulty that they are less certain.

The model should be at least inspectable (6). This may not allow students to alter its contents, and would therefore not provide control over their learner model or indirect control over the interaction, but it could offer greater opportunity to take control over learning through identification of the extent of their knowledge and potential problems (as long as the system allowed them some guidance of the interaction). A negotiated model would offer greater control, as the student could influence the model by argumentation. An editable model would give the greatest level of learner control.

A student must be able to initiate access (7) to their model to control it fully.

The student should have complete access to some sources (8) of input to their model. This applies in particular to the system's model and learner's own contributions. To help learners gain more control over their learning, it may be useful to have access to other sources. For example, an environment might allow the learner to view contents inferred by the system and also data contributed by peers. The student could inform the system of the information sources that should be regarded as most accurate to help shape the interaction. Against this is the possibility of others being more capable of making judgments about the student. Nevertheless, for the question of control, it is the learner's own control that is the issue. A different scenario might be a learner able to alter data supplied by another, if they believe it incorrect. For learner control this may be acceptable. However, for educational purposes it may not be appropriate to rely too heavily on the learner's views if they contrast with those of other people.

The learner should have complete control over the accessibility to other people (9), of their learner model, as well as to other programs using their learner model.

In order that a learner has control over their model, interaction and learning, they need full understanding of the effect of the model on personalisation (10).

Flexibility of access to the learner model should be complete (11).

Right to View the Model:

If learners should be allowed access to their data, as a right, this access should be complete (1). This includes representations of knowledge, difficulties, learning and social issues, preferences, and any other information (1).

The format of viewing is not prescribed, but learners should be able to access whatever details they wish - all, or selections (2). The latter issue of selecting data to view, is important to enable access in a manner which allows understandable presentation (information the learner wants, without being overwhelmed by additional data).

If opening the model as a right, the model should be viewable in the same (or similar) form to the underlying representations (3). This does not prevent other presentations being used in addition, to facilitate viewing and support learning. Moreover, if right to view implies right to understand, additional representations may be vital.

Since learners should have access to all data, this includes uncertainty (4). Similarly, access should be provided to all data held over time (5) - previous, current, future, if the system can predict that. The model need only be inspectable (6). Access must be able to be initiated by the learner (7).

Students should have access to complete information regarding sources of input (8). This applies to input from the system, explicitly from themselves, or other programs. In the case of input by peers - the student should know which data came from peers, but the privacy of the peer contributing data also needs to be protected. This is a difficult issue: whose right is greater? Ways around ethical problems might be to allow peers to contribute information only if they agree to their identity being revealed. However, this may stifle peer contributions; resulting in a decrease in the utility of learner models that can benefit from peer offerings. Whether the teacher's contributions should be named, is perhaps less controversial. However, does this violate teachers' rights? Conversely, if teachers are not identified, does this violate students' rights? Similar arguments apply where other people contribute information.

A learner should have complete control over access to their learner model by others (9). This is essential if considering open learner modelling from the perspective of the rights of individuals. This control applies to all potential accessors: the system, themselves, peers, teachers, other programs, and any other potential accessor.

The remaining aspects (10, 11) are less important for this goal.

3. An Example

The various aspects of open learner models are not usually described in sufficient detail to enable a reconstruction of Table 1 showing how some of the main open learner models to date, fit into the design space. We therefore use one of our own systems as an example.

Mr Collins [6] is a system with an open learner model for language learning, aimed at university level students. While several purposes of opening the model, and elements of openness, are relevant, only the most critical are given here as an illustration of the framework in use (Table 2). Mr Collins aims to improve the accuracy of learner modelling using a negotiated model, while also promoting reflection through the negotiation process. Thus the openness of the model is vital. A negotiated model is necessarily inspectable, and the negotiation process can be initiated by either learner or system. Access to sources of input to the model from both system and learner, is complete. Full access is available to data about knowledge, difficulties, other learning issues (e.g. language transfer), preferences (learning strategies). The above are useful for both purposes of opening the model described here.

Table 2. Mr Collins in SMILI☺

Open Learner Model	*Properties*	*Accuracy*	*Reflection*
0.Centrality of openness		X	X
1. Extent of model accessible	Complete	X	X
	Knowledge Difficulties	X X	X X
	Learning Preferences	X X	X X
2. Presentation	Textual	X	
	Overview Targeted All Details	 X	X X X
3. Similarity underlying	Different		X
5. Role of time	Previous Current Future	 X 	X X X
6. Access method	Inspectable Negotiated	X X	X X
7. Access initiative	Sys initiated Stu initiated	X X	X X
8. Access to sources of input	Complete	X	X
	System Self	X X	X X

Presentation is by text and tables - detailed text being important to ensure that the learner understands the model contents before trying to negotiate changes. The level of detail to support reflection is up to the learner. Given that promoting reflection is a key aim of the open learner model, it is important that the learner can understand the model contents. Thus the representation is different from the underlying Prolog. To achieve an accurate model it is important that the learner can access the current model. To prompt reflection the learner can also track their progress over time, both backwards and forwards to anticipated future states.

Using SMILI☺, contrasting Mr Collins with another system is quite easy - for example, the editable learner model for the SAM text editor (also for university students) [4]. While, like Mr Collins, several purposes and elements of openness are relevant, the main aims are to increase model accuracy by permitting the learner to directly change contents with which they disagree, thereby also allowing the learner to take greater control over the interaction. The two main 'purpose' columns would be *accuracy* and *control*. The main difference between the systems would be *access method* (editable rather than negotiated). Other differences include the *presentation format* (though similarly to Mr Collins, the format does differ from the underlying representation); and *role of time*. In both systems, the third most relevant column would be *planning/monitoring*, in particular the use of the open learner model to help students gauge their progress, and identify topics and concepts requiring more focussed effort.

4. Summary

The SMILI☺ Open Learner Modelling Framework provides a method to describe and analyse open learner models. The framework should also facilitate consideration of issues relevant to designing the openness of learner models in future adaptive learning environments. The framework also suggests dimensions of useful evaluations: for example, has the selection of the knowledge presented actually supported the goals of the openness? We expect that, for certain systems, some of the links between elements and purpose of modelling suggested as generally important, and those described as less crucial, may differ. The framework is intended to be flexible enough to allow for this currently, as unusual approaches to open learner modelling are described, and in the future as the field evolves.

References

[1] Self, J. (1999). Open Sesame?: Fifteen variations on the theme of openness in learning environments, *International Journal of AI in Education* 10, 1020-1029.

[2] Weber, G. & Brusilovsky, P. (2001). ELM-ART: An Adaptive Versatile System for Web-Based Instruction, *International Journal of AI in Education* 12(4), 351-384.

[3] Zapata-Rivera, J-D. & Greer, J.E. (2001). Externalising Learner Modelling Representations, *Proceedings of Workshop on External Representations of AIED,* AIED 2001, 71-76.

[4] Kay, J. (1997). Learner Know Thyself: Student Models to Give Learner Control and Responsibility, *International Conference on Computers in Education*, AACE, 17-24.

[5] Dimitrova, V. (2003). StyLE-OLM: Interactive Open Learner Modelling, *International Journal of AI in Education* 13(1), 35-78.

[6] Bull, S. & Pain, H. (1995). Did I say what I think I said, and do you agree with me? Inspecting and Questioning the Student Model, *Proceedings of AI in Education*, AACE, 501-508.

[7] Bull, S., Mangat, M., Mabbott, A., Abu Issa, A.S. & Marsh, J. (2005). Reactions to Inspectable Learner Models: Seven Year Olds to University Students, to appear in *Proceedings of Workshop on Learner Modelling for Reflection*, International Conference on Artificial Intelligence in Education, Amsterdam, 2005.

Enhancing Metacognitive Skills through the use of a Group Open Learner Model based on the Zone of Proximal Development

Nilubon TONGCHAI and Paul BRNA

The SCRE Centre, University of Glasgow
11 Eldon Street, Glasgow, G3 6NH
nilubon.tongchai@scre.ac.uk; paul.brna@scre.ac.uk

Abstract: Collaborative Learning is seen as a good way to encourage peers to learn and to teach each other whereas Open Learner Modelling can help learners to enhance their metacognitive skills and their understanding using high-level indicators to monitor, and represent, the state of their learning. In this work we aim to develop a learning environment that encourages students to obtain an advantage from both Collaborative Learning and Open Learner Modelling. We then seek to determine the benefits of Collaborative Learning with a scrutable Group Learner Model[1] by examining the learning gains when compared with the case in which no Group Learner Model is available.

Introduction

Collaborative Learning is seen as a good way to encourage peers to learn and to teach each other whereas Open Learner Modelling can help learners to improve their performance and their understanding using high-level indicators to monitor, and represent, the state of their learning. This research seeks to apply both concepts of Collaborative Learning and Open Learner Modelling.

Why collaborative learning? Following Vygotsky who argued that learning had a strong social dimension, we believe that learners can often better improve their knowledge while learning with peers than learning individually. In this work, we exploit the notion of Vygotsky's Zone of Proximal Development, defined as "the distance between the actual developmental level as determined by independent problem solving and the level of potential development as determined through problem solving under adult guidance or in collaboration with more capable peers" [2, p.86].

An Open Learner Model is often considered to be an aid to reflection. Bull defines an Open Learner Model (OLM) as a student model which is designed to help learners understand what they have learned more effectively[3]. This kind of model allows the learner to inspect, and sometimes challenge, beliefs recorded in the user model which encourages the learner to think more deeply or extensively about their understanding.

Group Open Learner Model emerges from the merging of a ‘Group Model’ and an ‘Open Learner Model’. An ‘Open Learner Model’ is simply thought of as an aid to reflection while a ‘Group Model’ is a more complex concept. While there are many works that use a group model, there are few that can define the OLM in a way that differentiates clearly between the emerging properties of the group and the properties of the individuals involved. For our work, we especially need to define what exactly the group model is, how it works, and precisely what the model includes. From Paiva [4], a group model is considered as 'a way of capturing the aspects that identify a group as a whole' and it may include group beliefs, group actions, group goals, group misconceptions, differences between individual and group conflicts.

Less has been done with Group Open Learner Models (GOLMs) though Zapata-Rivera and Greer[5] found that students could be very confused when seeking to understand their GOLM. However, this GOLM was developed by a group of students working together with a single instance of Zapata-Rivera's ViSMod system. The issue of the GOLM is taken up again later.

1. Research Problems

During past decades, many tools and methodologies have been designed to support Collaborative Learning interaction. The focus of this research topic is shifted from 'studying group characteristics and product', which contain many unpredictable factors, to 'studying group process' in the nineties. Jermann, Soller et al [6] introduced the idea of the 'Collaboration Management Cycle', which consists of four phases: Collect interaction data, Construct a model of interaction, Compare the current state of interaction to the desired state and Advise/Guide the interaction. This cycle provides a conceptual framework for managing collaborative interaction. In their view, all the four phases above are covered by three computer-based support options: Mirroring tools, Metacognitive tools and Guiding Systems.

When someone learns a topic, either on their own or with a friend, they may need to know how well they performed on that particular task. In the classroom, the teacher may give some information such as a score or some suggestion about performance on the task[1]. An Open Learner Model is considered to be an aid to reflection insofar as it can convey - directly or indirectly[2] - such information, and provokes the learner to think about the truth or falsity of the information conveyed, and in doing this, reflects upon a number of issues including perhaps that of how their learning is progressing.

An Open Learner Model is seen as the model that reflects back to the learner information that lets them know how well they are performing particular tasks or how well they understand some concept. From the information provided, learners then become aware of their knowledge and decide what should do next. The generally held belief amongst researchers is that it is possible to improve learners' knowledge by showing them their learner model [7-9]. To investigate this belief, concepts of 'Theory of Mind' and 'Meta-Cognition' are considered as crucial factors to understand how OLMs help improve knowledge (skills).

1.1 Theory of Mind, Metacognition and Metacognitive Skills

One definition of 'Theory of Mind' is as 'a specific cognitive ability to understand others as intentional agents to interpret their mind in terms of theoretical concept of intentional states such as beliefs and desires'[10]. In short, theory of mind is 'an awareness and understanding of mental processes'. For example when a learner performs a specific task

[1] This might be done in absolute or relative terms e.g. you got 7/10 or you did better than the average.
[2] By indirect, we mean that the information may not be explicit but can be inferred from the information provided.

and the system reflects back the score or some other information, the way that learner try to understand what the system reflects back is what the system believes about a learner's knowledge and skills.

One form of 'Metacognition' is often simply defined as 'thinking about thinking'[11]. However defining Metacognition is not simple because there is still much debate over what metacognition means for a couple of decades. Defining by Wilson[12, p.14], "Metacognition is the knowledge and awareness one has of their own thinking processes and strategies and the ability to evaluate and regulate one's own thinking processes".

According to Flavell [13] metacognition consists of both metacognitive knowledge and metacognitive experience or regulation. Metacognitive knowledge is briefly stated to acquire knowledge about cognitive process and how to use knowledge to control the cognitive process. Flavell divided metacognitive knowledge into three categories: knowledge of person variables, task variables and strategy variables. Knowledge of person variables contains information about how well a particular person learned and processed information while knowledge about task variables considers the nature of the task to provide a suitable environment for the most productive results (e.g. reading a physics book is harder to understand than reading a novel so more time should be provided for this physics task). Knowledge strategy variables are concerned with when and where appropriate strategies are being applied.

Metacognitive experience involves the uses of Metacognitive regulation to control cognitive activities to ensure that the cognitive goal has been met. For example, after reading a lesson asking oneself what one has got from the lesson. If the question cannot be answered, then go back to the lesson again and at the same time determine what else can be done to ensure that that lesson has been understood.

Schraw [14, p.121] has develop a regulatory checklists that student can use to monitor their own metacognitive control. There are three groups of checklists: Developing a plan of action, Monitoring the plan (being aware of everything that has been done by oneself), and Evaluating the plan. The following is an initial adaptation of Schraw's checklists for the group learning[3].

1. Checklists for developing a plan of action (Before performing a task)
   - How much does my peer know?
   - How much is our prior knowledge?
   - How can I get my peer to help me?
   - What should we do first?
   - How much time is needed to complete this task?

2. Checklists for Monitoring a plan of action (During performing a task)
   - How are we doing?
   - Can I make a group contribution?
   - How should we proceed?
   - What do we need to do if neither of us understand?

3. Checklists for Evaluating a plan of action (After performing a task)
   - How well did we do?
   - Did we do more or less well than what we had expected?
   - What could I have done differently?
   - How well I have helped peer to learn better?

---

[3] These checklists suggest a way of evaluating metacognitive activity

If learners have such a Metacognitive experience, we assume that they will have self-awareness of what they know and what they do not know, and what they should do to complete the given task. In Collaborative Learning, there is a need not only to understand themselves but also to understand others which motivates our concern to include notions of Theory of Mind. Thus knowing how the group is doing and reflecting upon the Group Open Learner Model, the learner also needs to understand themselves so that they can determine their weak and strong points. At the same time, they may need to take into account the knowledge of their peers and the potential for their peers to help them.

1.2. Collaborative Learning and Zone of Proximal Development

Collaborative Learning is interpreted here in two distinct ways - the way that learners help each other in a group and the way that a teacher or a learning system helps the student to gain a better understanding. Teaching collaboratively helps learners to learn skills and ideas initially in their ZPD which is why "collaborative teaching" is important. Murray and Arroyo [15] implemented a learner model to support the concept of ZPD – their work illustrated that the student who masters material collaboratively today can master it individually tomorrow.

Related to the idea of ZPD is that everyone may be in a different state of learning in a group. Hence with a user model, either a personal or a group model, it is possible to individualise the level of knowledge to provide a suitable degree of reflection. However for developing more efficient collaborative learning, empirical studies have changed the focus from 'establishing parameters' to trying to understand the role which such variables play in mediating interaction [16].

There are many systems that are used for Collaborative Learning, some of which refer to the concept of ZPD, some reflect back the learner model to an individual student and a very few use a GOLM but how many of them contain both concepts of reflecting back group knowledge and explicit use of the notion of the ZPD? Six systems have been selected as representative of the state of the art; these are compared.

Table 1. The comparison of systems to represent concept of ZPD, individual and group learner model

System's name	References	Did they use the ZPD concept explicitly[4]?	Did they reflect back to individual learner?	Did they reflect back the group learner model?
ViSMod[5]	[5]	No	Yes	No
ECOLAB	[7]	Yes	Yes	No
ICLS	[17]	No	Yes	Yes
PairSM	[18]	Yes	Yes	No
STyLE-OLM	[9]	No	Yes	No
Mr.Collins	[19]	No	Yes	No

1.3 Group Learner Model

Group Open Learner Model emerges from the merging of a 'Group Model' and an 'Open Learner Model'. Paiva [4] described two scenarios which represent her notion of a group

[4] We mean that internally there is a model of the learner which represents ZPD in some direct ways.

[5] Another version of ViSMod describes some works with a Group Model but not the kind that we are interested in.

model. The first scenario is to combine multiple individual models for the possible peer group (this notion is presented by Hoppe[20]). The second scenario is about learners who interact with the collaborative environment for which all of these properties should be considered: a shared-task space, a communication space, authorisation to see the communication, a domain model and an individual-task space.

According to Table 1, ViSMod, STyLE-OLM and Mr.Collins are systems that reflect back only to individual learners whereas PairSM and Ecolab use both the concept of ZPD and reflect back the model to each learner. ICLS reflects back both individual and Group Learner Models. However none of the systems above uses all of the concepts - namely, ZPD and reflecting back the individual and group models. In this paper, the ideas of the system that utilises both the concept of ZPD and reflecting back the Group Learner Model are illustrated. The first question is why we want to utilise a Group Model and the second is how are we going to generate a Group Model?

1.3.1 How are we going to generate a Group Model?

Most people see the Group Model as some kind of addition of individual models. Hoppe[20] combined multiple individual learner models with the aim of forming more effective peer groups though Paiva [4] looked for something potentially better by combining the concept of a group model with an individual learner model to construct a basic framework for models in collaborative situations. However PairSM, a model that applied a simple picture and a set theory equation to illustrate the Group Learner Model, seems to be interesting because it considers a Group Learner Model together with the notion of the ZPD even though the group model comes from a simple combination of the individual learner models. The explanation above can express a Group Model as an equation $SM\text{-}S1S2 = SM1 \cup SM2 \cup SM\ S1\&S2$,which SM represents the knowledge of an individual learner, and SM S1&S2 represents knowledge that the two can display only when working together. The group model in this work will borrow ideas from both Paiva and PairSM to generate the group model for Collaborative Learning with considering to ZPD concept.

1.3.2 How are we going to represent the model?

There are many possible ways such as text and graphical form that we could represent the learner model. STyLE-OLM [9]uses a diagrammatic form of conceptual graph to represent the learner model and the text form for an interaction model. Moreover users can swap between learning mode and interaction mode to see what they have done in the past. ViSMod [5] uses different colours and link sizes and nodes to indicate the level of knowledge for particular learners for each concept. This should help learners quickly distinguish how well they perform for each concept. In our work, we will use a text form to represent the interaction model, while a graphical form will be used to indicate the level of knowledge for both group and individual learners.

1.3.3 How are we going to manage the interaction?

STyLE-OLM, and ICLS use different means of tagging individual moves in the interaction. STyLE-OLM uses the notion of a dialogue game for interactive communication between a learner and the system, while the open learner model concept allows student to inspect and negotiate their own model. Mr.Collins aims at improving learning through promoting reflection by giving a chance to both students and a system to defend their beliefs using the difference of confidence in beliefs between the learner and the system. Whether learners can challenge and negotiate models through menu, changing models ultimately depends on the rules programmed into the system.

ICLS (Intelligent Collaborative Learning System) provides a good example of the use of sentence openers. This emphasises the role of communicative interaction. The ICLS system classifies groups of sentence openers, helping the group know how well they perform. In our work, we borrow the idea of dialogue game and sentence opener for the communication interaction and the level of confidence for their beliefs to generate the learner model. In our research we focus on a Group Open Learner Model for Collaborative Learning. The group model will borrow ideas from Paiva and Bull's PairSM to generate the group model while taking the notion of the ZPD into account. Dialogue game and sentence openers will be used for communication interaction whereas it is planned to use a pie-chart and text as mirroring tools to represent the learner's beliefs and knowledge.

2. Evaluation

It is currently envisaged that two conditions for learning with a peer are compared: 'can see the group model' and 'cannot see the group model' using a bar-chart and some explanation to represent the information of each sub-concept that the group performs. The hypothesis is that learning with a peer and seeing the information reflected back as a group model will help the learner get a higher score than not seeing the group model.

The learner model - either group or individual - contains elements as a member of set for each sub-concept. There are two major types of information that are represented in the learner model: ‘Experience’ and ‘Inexperience’ value. In this system, the‘Experience’ value contains one of these three values: ‘K’ as 'Known that a sub-concept is used correctly', ‘M’ as 'MayKnown for a sub-concept is sometimes used correctly', and ‘N’ as 'Notknown for a sub-concept that is used incorrectly'. Values represent the performance of using a sub-concept for previous tasks undertaken by particular learners - while the ‘Inexperience’ value for a sub-concept represents the situation that the learner has not tried to perform a task involving that concept before (as far as the system knows).

There are two types of group models: GLM (Group Learner Model) and Ideal GLM (Ideal Group Learner Model -see Figure 1).

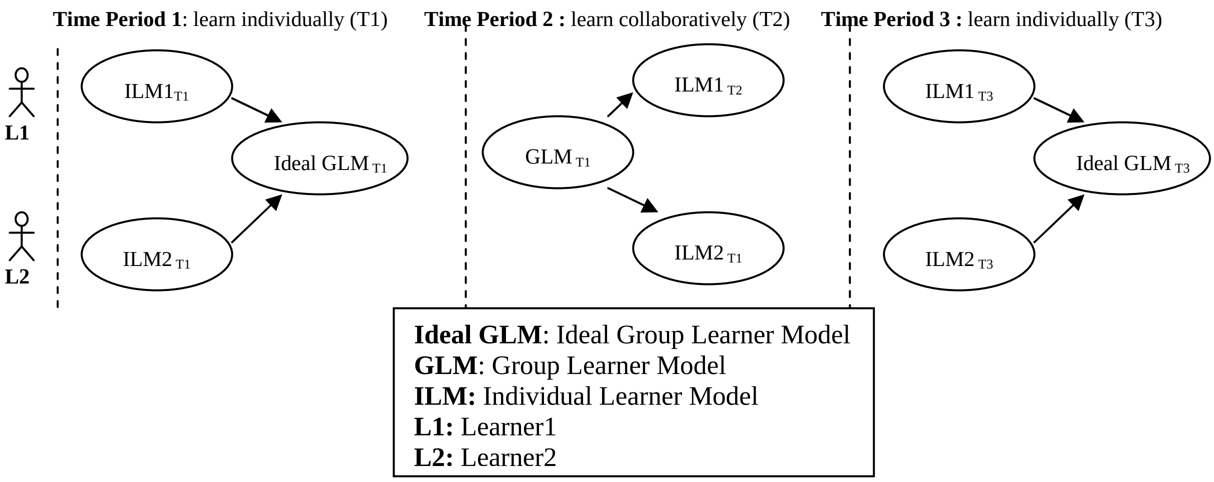


Figure 1. A Group Model Diagram

To fulfil that aim, we have decided to use two students learning together[6]. Each student can choose their peer as they wish from the list that the system provides. Before starting to learn with a peer, the learner registers with the system and takes a pre-test. The learner's information is kept individually for use in the future.

As seen in Figure 1, there are two learners L1 and L2 who have decided to learner together. Firstly they perform the task individually during Time Period 1 and submit their answer to the system at the end or the period. After that the results of learning are checked, scored, and are reflected back to learners as the Ideal GLM. The Ideal GLM is expected to promote self assessment and self-awareness (at least). Encouraging Metacognition, learners would be supposed to assess both their own knowledge and their peers' knowledge from the provided information.

The Ideal GLM uses a model merging algorithm to derive a group model which is ideal in the sense that the merging of models is intended to show the potential of the group (prior to further learning taking place). In a group model, we present the values that calculate from the difference between the Ideal GLM and GLM in terms of bar-charts with some explanatory information. If learners see these details and perform better than learners who cannot see this information, we may be able to conclude that a group model[7] is effective for collaborative learning.

During Time Period 2, learner are provided with the environment for the group task which allows interaction with both peer and the system by using templates provided for generating the dialogue. These dialogues rely on the concept of a dialogue move so that the system can categorise what learners try to say to each other, and will be used to estimate what learners understand of that particular task. Each dialogue move that learners use will contribute a score which affects the assessment for each concept of the group model. The approach will rely technically on the use of fuzzy logic.

After finished a task, the system will reflect back the information of the group performance using GLM. At this stage, the result of an individual model ($LM1_{T1}$ and $LM2_{T1}$) from T1, which represent the actual knowledge of particular learners, will be compared to the result of GLM_{T2} from T2. Differences of results are expected to be a potential performance of these learners and are kept in $LM1_{T2}$ and $LM2_{T2}$.

A simplified version of the ZPD is ability for doing something that you cannot do on your own but you can with others. The ability that learner can do something without any help sometimes is known as 'actual performance'. Whilst 'potential performance' represents the ability that with some help, one can complete tasks. In order to turn a potential performance into an actual performance, learners should repeat similar tasks individually as seen in Time Period 3 (Figure 1) after doing them collaboratively in Time Period 2. This time the information of each individual learner at each specific times is used to compare and calculate showing that learner can improve their knowledge and performance by collaboratively learning using this system. However no one can guarantee that particular learners will always succeed on the similar task again when doing it individually.

A prototype will be built to demonstrate the working of the model and it is expected to use fuzzy logic for dealing with the uncertainty in such a model. After the model has been developed further, the approach above will be implemented, tested and revised prior to developing the model used for the final study with learners. A repeated measure design within subject will be used to compare the result of learning to show that collaborative learning with the Group Open Learner Model is better than without the Group Open Learner Model.

[6] i.e. the pair will be regarded as a group, a simplifying assumption that we will seek to lift later.

[7] Note that this is a strong statement - the learners will *not* be shown their individual model. We are currently constructing a theoretical account of how this may work.

3. Conclusion

Collaborative learning is a good way to encourage peers to learn and to teach each other whereas Open Learner Modelling gives learners an opportunity to inspect or sometimes challenge[8] their user model to make it more accurate and to learn from this process. The work described here aims to encourage students to obtain an advantage from both collaborative learning and the use of an Open Learner Model to try to prove that the result of collaborative learning using a Group Open Learner Model helps them get a higher score than when unable to inspect the group model. Since we also want to determine whether such an experience also contributes to the enhancement of the learners' metacognitive skills, we are currently considering how to extend the experimental design. After this work is done, further work will concentrate on 'In what ways is a Group Learner Model better than an individual Learner Model?'

References

[1] Kay, J. Stereotypes, Student Models and Scrutability. In *ITS 2000* (pp.19-30). 2000: Springer-Verlag Berlin Heidelberg.

[2] Vygotsky, L., *Mind in Society: The Development of Higher Psychological Processes*. 1978: Cambridge, MA: Harvard University Press.

[3] Bull, S. and T. Nghiem. Helping Learners to Understand Themselves with a Learner Model Open to Students, Peers and Instructors. In *Proceedings of Workshop on Individual and Group Modelling Methods that help Learners Understand Themselves, ITS 2002* (pp.5-13). 2002.

[4] Paiva, A. Learner modelling for collaborative learning environments. In B. du Boulay, & R. Mizoguchi (Eds.) *Artificial Intelligence in Education* (pp. 215-222). 1997. Amsterdam: IOS Press.

[5] Zapata-Rivera, J.-D. and J. Greer. Externalising Learner Modelling Representations. In *AI-ED 2001 Workshop External Representations in AIED: Multiple Forms and Multiple Roles* (pp.71-76). 2001. San Antonio, Texas.

[6] Jermann, P., A. Soller, and M. Muehlenbrock. From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning. In *Proceedings of the First European Conference on Computer-Supported Collaborative Learning* (pp.324-331). 2001. Maastricht, The Netherlands.

[7] Luckin, R. and B. du Boulay, Ecolab: The Development and Evaluation of a Vygotskian Design Framework. *International Journal of Artificial Intelligence in Education*, 1999. **10**: 198-220.

[8] Zapata-Rivera, J.-D. and J.E. Greer. Interacting with Inspectable Bayesian Student Models. *International Journal of Artificial Intelligence in Education*, 2004. **14**: 127-163.

[9] Dimitrova, V. STyLE-OLM Interactive Open Learner Modelling. *International Journal of Artificial Intelligence in Education*, 2003. **13**: 35-78.

[10] Davidson, D. Theories of meaning and learnable languages. In *Inquiries into truth and Interpretation* (pp.3-15). 1984, Oxford: Oxford University Press.

[11] Livingston, J.A., *Metacognition : An Overview*. 1997.

[12] Wilson, J., Assessing metacognition: legitimising metacognition as a teaching goal. *Reflect*, 1998. **4**(1): 14-20.

[13] Flavell, J.H., Metacognition and cognitive monitoring: A new area of cognitive-developmental inquiry. *American Psychologist*, 1979. **34**: 906-911.

[14] Schraw, G. Promoting general metacognitive awareness. *Instructional Science*, 1998. **26**(1-2): 113-125.

[15] Murray, T. and I. Arroyo. Toward Measuring and Maintaining the Zone of Proximal Development in Adaptive Instructional Systems. In S.A. Cerri, G. Gouardères & F. Paraguaçu (Eds.) *Proceedings of the Sixth International Conference on Intelligent Tutoring Systems* (pp.749-758). Berlin: Springer-Verlag. 2002.

[16] Dillenbourg, P., et al. The evolution of research on collaborative learning. In *Learning in Humans and Machine: Towards an interdisciplinary learning science* (pp.189-211). 1996: Oxford: Elsevier.

[8] Currently, we are planning to allow students to inspect but not change their GOLM.

[17] Soller, A., et al. What Makes Peer Interaction Effective? Modeling Effective Communication in an Intelligent CSCL. In *The 1999 AAAI Fall Symposium: Psychological Models of Communication in Collaborative Systems* (pp.116-123). 1999. Cape Cod, MA.

[18] Bull, S. and M. Smith. A Pair of Student Models to Encourage Collaboration. In A. Jameson, C. Paris & C. Tasso (Eds.) *Proceedings of the Sixth International Conference on User Modeling* (pp.339-341). Berlin: Springer-Verlag.

[19] Bull, S., H. Pain, and P. Brna, Mr. Collins: A Collaboratively Constructed, Inspectable Student Model for Intelligent Computer Assisted Language Learning, *Instructional Science*, 1995. **23**: 65 - 87.

[20] Hoppe, H.U. *The use of multiple student modelling to parameterise group learning*. In *Artificial Intelligence*. 1995: AACE. 234-241.

Dimensions of Transparency in Open Learner Models[1]

Steven Tanimoto [2]
Department of Computer Science and Engineering
University of Washington, Seattle, WA, 98195, USA.

Abstract. The design of learner models that are open to the student's perusal is challenging, because a variety of competing objectives must be reconciled: comprehensiveness versus comprehensibility, and student control versus model validity. This paper suggests one approach to meeting the challenge that begins by identifying three dimensions of transparency in learner models. These provide a framework in which to design appropriate views of the data and limited controls, that best satisfy the requirements for the model. Part of one solution is to logically and physically distribute the records that comprise the model. The current status of learner modeling in the INFACT system is described.

Keywords. learner model, student model, user model, transcript, computer-based learning environments, transparency, tutoring system, open learner model, metacognition, dimension.

1. Introduction

A *learner model* is a computer-based data management component or system that contains information about a person's learning activity. It typically forms a part of a larger system such as a learning management system or an intelligent tutoring system. An *open* learner model is one with specific provisions for the learner to have one or more views of the information in the model[2]. Furthermore, these views are relatively comprehensive, hiding relatively little from the learner. An open model can be contrasted with a closed model in which the student has no direct view of the model's contents.

An important potential advantage of open learner models is that they may encourage valuable metacognitive activity and thereby help a learner to learn more effectively[7,2]. Another possible benefit is that they may permit modelling errors to be detected more readily so that they can then be corrected[3,1].

Influences promoting open learner models include the following: the trend toward accountability in artificially intelligent systems, exemplified by explanation features in expert systems of the 1970s and 1980s; movements to create standards for learning technology; efforts to help learners take better advantage of the growing array of resources on the World Wide Web; gaining students' trust[14]; and recognition by psychologists of the value of metacognition by students. Technology to support proactive learners[9] and more detailed transcripts for learner-to-resource matchmaking[10,11] also call out for open learner models.

There are several challenges in designing open learner models. These challenges arise primarily from conflicting demands for which good compromises must be found. One axis of conflict is openness versus confusion; learner models may be complex and contain components that are difficult to explain, and presenting their details to unsophisticated learners may not only confuse them but upset

[1]Research supported in part by the US National Science Foundation under grant EIA-0121345.

[2]Correspondence to: Steven Tanimoto, Box 352350, Department of Computer Science and Engineering, University of Washington, Seattle, WA 98195, USA. Tel.: +1 206 543 4848; Fax: +1 206 543 2969; E-mail: tanimoto@cs.washington.edu.

them. Another challenge for open modelling is keeping the model valid. If a model shows a learner's warts, he or she may be tempted to "fix" it by hacking into it to alter or remove unflattering parts.

To meet the challenges, three aspects of transparent models are identified. These form a basis that spans a space of abstract designs for learner models. A good design for a sophisticated open learner model will tend to be at an extreme point of the space.

2. Transparency in Learner Models

A *transparent* system is one that makes some of its inner workings visible to the user[12]. It can be contrasted with a "black box" that accepts inputs and returns outputs but hides the transformation mechanism. Transparency is often a desirable feature in software systems, because it can reveal to users how the system works, helping to engender trust, permit error detection, and foster learning about how software systems work.

Let us consider the application of transparency to some learner models. To start, let's look at some relatively simple kinds of models: overlay models and locus-within-strand models. Then, we'll examine more challenging ones such as facet-based models and student achievement scores computed using complex inference processes.

2.1. Student-Understandable Learner Models

An overlay model typically takes advantage of a taxonomy of skills and concepts within an academic subject and represents the student's state of knowledge using a set of numerical scores, with one score per skill or concept. It is easy to make such a model transparent by simply providing to the learner, either on demand, or through the initiative of the program, a list of skills and concepts with the student's current scores for them alongside them. If the list is organized hierarchically, then section (or other subdivision units) scores and a total score can also be given. Such a model and its view do not communicate to the learner the reason for any of the basic scores. The learner might assume, for example, that the scores are the results of answering questions correctly or incorrectly.

A strand-based learner model organizes skills and concepts into linear sequences called strands. A strand typically consists of a progression of subtopics in which one subtopic may have prerequisite subtopics that appear earlier in the progression. A strand-based tutoring system might start a session with a student at the beginning of a strand and only allow the student to progress to the next subtopic in the strand when its preceding topics have been mastered sufficiently. The student's state within the strand is considered to be completely represented by the index of the last subtopic mastered (the student's "locus" within the strand.). This kind of learner model is somewhat like a coarse-grained overlay model, because a student may have separate loci in a number of different strands at the same time. However, the meaning of the locus is somewhat different from the meaning of the overlay model's mastery value for a single topic.

It is easy to open up a strand-based learner model by identifying the subtopics in the strand and showing the learner the locus, either in real-time, on-demand, or at the end of a session.

If learner models were always this simple, it would be fairly easy, in principle, to keep them transparent and fully open to learners. There would still be issues to resolve regarding best modes and styles of presentation, though. Complex models, however, pose significant challenges to designers of open learner models.

2.2. Difficult Learner Models for Students

Let's consider two somewhat more problematical kinds of learner models. One is called a "facet-based" model, and it is based on the work by Minstrell to categorize common misconceptions in physics[8]. Like an overlay model, a facet-based model contains a construct for each skill, concept or subtopic within an academic subject. However, the construct is not simply a number that represents degree of mastery but a probability distribution over a set of conceptions and misconceptions. In the simplest case of such models, the probability values are restricted to being 0 or 1, so that only one facet (conception or misconception) is indicated for a given subtopic for a given student at a given time.

Making a facet-based learner model transparent to the student is problematical for two reasons. First, facets are defined in pedagogical terms as well as subject-area terms, and students cannot be expected to understand either of these sublanguages; generally, students are not teachers, and students are not experts in the subjects they are currently studying. Second, the probability values assigned to the various conceptions and misconceptions may be the result of processes that are incomprehensible to the student. Achieving a useful kind of transparency for facet-based learner models therefore requires not only a means to reveal the names and values of model variables but an interpretive mechanism that translates the information from a pedagogical perspective to a learner's perspective. Without such a mechanism, an open facet-based learner model would at least have to have accompanying documentation that says as much as "students are not expected to completely understand the classifications of their states of knowledge of particular topics or the inference processes by which those classifications are made."

A kind of learner model complexity different from the explicit incorporation of misconceptions is complexity which occurs in the inference processes that produce model values. A good example of an inference process that is difficult for students to understand is Latent Semantic Analysis (LSA), which is a technique that uses numerical matrix computations to compare textual documents. It has been used for essay grading and student written-answer classification. LSA typically is applied by converting a student's essay into a vector of word occurrence counts. That vector is compared with vectors representing, say, good and bad essays, or essays in various topical or stylistic categories. Providing an accurate description of this process to, say, an English major, would probably not be very helpful.

2.3. Gaming the System

There is another, very different, problem for achieving transparency with a learner model based on a technique such as LSA. Although the numerical method itself is not something that a student could understand without having studied linear algebra, there are properties of LSA that a student can easily understand that teachers and testing agencies may wish to keep hidden from students. LSA-based scoring ignores word order in the input and bases its results only on the frequencies of occurrence of the words. A student who knows this might be tempted to game the grading system, find out what sorts of words will be required to be used for a particular kind of essay, and submit gobbledygook on a writing assignment that nonetheless satisfies the LSA-based assessment system. This is, of course, not specifically a problem for learner models, but for educational assessment in general. However, learner models are products of educational assessment, and making them open may expose parts of the assessment process that will no longer work properly if exposed. An open learner model should not be one that says, "Here's how to fool the system."

3. Validity in Learner Models

In the preceding section, we developed the issue of transparency versus learner confusion — some models may contain information that students cannot be expected to understand. Now let us consider another challenge for transparency in learner models: maintaining the validity of the information in the model.

There are two threats to the validity of data in the model. The more serious of these is that students, once aware of what their learner models represent, and perhaps inadvertently empowered to edit those models, make changes to the models' data that corrupt the models and that make them unfit for the purposes for which they were designed, e.g., to improve learning. The other threat may arise as a design bias.

3.1. The Risk of Tampering

Transparency has some obvious benefits. However, as a means of raising students' awareness of their weaknesses as well as their strengths, transparency can beget a desire to control. The extent to which a learner model becomes a representation of success or failure in any learning domain can lead to a lessor or greater wish on the part of the learner to change the representation (rather than the phenomenon it represents). Just as some students have hacked into school computer systems to change their grades, some students may attempt to gain editing access to details of their learner models and change them, instead of trying to change their own knowledge.

Transparency without appropriate validity protections would pose a substantial risk. The modern-day epidemic of cheating in school in countries such as the United States could be the downfall of open learner models.

3.2. The Risk of Design Bias

The other threat is unlikely to be catastrophic, but may have a subtly negative influence on the effectiveness of the models in guiding educational software systems. That threat is that designers, avoiding the incorporation of model components that are problematical for transparency, weaken the models' pedagogical value. For example, a designer might decide that it is too difficult to have the software explain to the learner what a particular facet assessment means and therefore decide that facet-based model components should not be included in the model. The components of the model may tend to be limited to lowest-common-denominator constructs: those that can easily be understood by all members of the expected learner population. Constructs with any degree of pedagogical sophistication would be avoided.

In the worst case, the model would consist of nothing but scores on tests and questionnaires, and would not contain any diagnostic results involving expressions of uncertainty.

4. Dimensions of Transparency

The fundamental challenge of designing transparent systems is answering the question of what to show and how to show it. The following three questions about this challenge lead us to corresponding "dimensions" of transparency:

- How much of the learner model is made available to the learner? The answer, which we could somewhat simplistically represent as a percentage, lies along an axis that we can call the *quantitative dimension* (or alternatively it could be called the "data dimension.")
- How much support is provided in explaining and interpreting for the student the learner-model data that is made available? This is the *interpretive dimension.*
- To what extent are authentication facilities visibly integrated into the model? This is the *validation dimension*.

In each of these dimensions the designers of a learner model may strive, through an appropriate mechanism, to achieve a balance between the needs of individual students and the need for consistent, reliable systems.

4.1. Quantitative Dimension

The question of what data within the model is made available can be considered as an elaboration on the question of how much is made available. Answering the *what data* question is clearly necessary in the design of a learner model, but it appears to be less fundamental, from a philosophical point of view, than the question of whether the model will be 100 percent open or not, or whether it will be open at all.

4.2. Interpretive Dimension

The interpretive dimension calls attention to the fact that nontrivial software development may be required to make complex models understandable by students. There is probably no limit on the extent to which explanation facilities might be taken in order to help students appreciate pedagogical judgments represented in the learner model.

There may appear to be some ambiguity between these first two dimensions, because an interpretation may call upon additional information that might seem to be a part of the model but is not explicitly recorded there. To explain a diagnosis (“student has misconception 2 on the topic of additive color mixing"), some context must be provided, which seems to be increasing the quantity of information disclosed (first dimension). However, if someone made a list of the items in the model and then chose one item to portray, one could argue that any interpretative information presented along with this item is attributable to this item and not the other items in the model. Thus the interpretive dimension can be kept distinct from the quantitative one.

Hansen and McCalla[6] have argued for “active learner models" that are constructed on-demand for particular purposes. With an active model, if a student wanted to know something about herself, she might query the model, and it would synthesize an explanation for her from the more-or-less raw evidence it encapsulates. It is more difficult to place such an active model on the interpretive dimension, because in a sense, an active model is itself a sort of interpretive process, and to interpret it would be to interpret and interpreter. In order to make such an active model transparent, it then becomes necessary to reveal the mechanism that generates the new model components.

4.3. Validation Dimension

The validation dimension arises directly in response to the challenge of maintaining model validity while putting it directly into the hands of students and opening it up. A learner model that falls at the high end of this axis has explicit validation facilities built into it in a way that both the learner and any agents accessing the model can see and activate. Much as a job resume lists personal or job references that can be contacted for information about the applicant, validation links in the open learner model identify agents such as web servers that are prepared to corroborate or elaborate upon elements in the student’s copy of the model. It is important not only that these exist in the model, but that they be visible to the learner. The transparency of the validation facilities can help the learner to put faith in the model and encourage the learner to maintain the integrity of the model.

One approach that addresses the validity issue in a novel way is the “negotiated learner model"[1]. With this method, a student may inspect the system-generated part of her model and try to change it through a kind of negotiation process. If the system refuses to change its evaluation on the basis of new evidence provided by the student, the student may still succeed in registering her disagreement in the model and thereby recording the objection. This by itself doesn’t prevent hacking, but it may reduce the temptation to edit values inappropriately by providing a legitimate “channel” for objections.

In addition to the issue of whether or not parts of a model may have been forged, in some situations a model could be incomplete; it could be a partial copy of the full model, or it could be a new fragment that has not yet been joined to or synchronized with its main model[4,11]. The possibility that alternative versions of a learner model may exist means that additional identifying information is required to specify, as clearly as possible, the relationship between the given version of the model and a reference version of the model.

5. Learner Models in INFACT

INFACT is a suite of tools for online communication, construction, and assessment[13]. At its center is a database containing evidence of student learning and evaluations of learning. The evidence consists of students’ textual discussion postings, graphical sketches (made with an online applet), and user-interface events posted by tools for computer programming and image processing. The assessments consist of “facet-assessment records” each of which comprises an evidence identifier, a facet (conception or misconception) identifier, a student identifier, an assessing teacher or agent identifier, a timestamp, and a certainty value. All the facet assessment records for a given student, together with the evidence pointed to by these records, constitutes a learner model.

Learner models in INFACT have some transparency features but are not yet very open. Rating them roughly on each of the three dimensions gives an abstract idea of what kind of system we have—*Quantitative: medium; interpretive: medium; validation: medium*. With regard to the quantitative aspect, facet assessments are, by default, private to the teacher at this time, although most of the evidence on which they are based (messages and sketches) remains available to the students. INFACT provides two interpretive mechanisms that teachers can configure to give instant feedback or delayed feedback to students. Configuration requires a significant effort, because rules and/or Bayes nets must be authored (using INFACT’s assessment development tools) for each facet within the topic being assessed.

Along the validation dimension, INFACT provides two features that contribute to maintaining the validity of each learner model. One of these is simply that all models reside on a protected server, and students cannot directly edit a model. The other is that each facet assessment record includes an assessor ID field, and thus a human (if the assessor was a person) could theoretically be asked to verify an assessment description that happened to be in doubt, or an agent (if the assessment was performed by software) could have its rules inspected by a teacher to check for consistency with the assessment.

In the future, we wish to add new transparency features to INFACT, both in order to evaluate their effects on learning, and to provide more options to teachers and students in organizing the online learning experience.

6. Discussion

Good strategies are needed for designing an open modeling system to take the three dimensions into account. Here are some possible aspects of a strategy.

On the quantitative dimension, the question of what parts of the model to show or not show can be divided into four parts: the easy and innocuous, the easy and more valuable, the harder but meaningful, and the inappropriate. The easy and innocuous material consists of the raw evidence that is collected in the model. If a student wants to see this, it can easily be shown, although some of it might not be very intelligible. The possible benefits to the student of offering this view include a feeling that the system is being open, the possibility of increased trust, and motivation to perform better due to the knowledge of being watched.

The easy and more valuable material consist of items that can be directly shared with students or shared with only a small amount of interpretation. If a student achieves a notable milestone in the course of a lesson, as this fact is recorded in the model, it can be shared with the student as it is recorded. Also in this category are skill meter readings[5]; they provide displays derived from values in the model using a relatively small amount of interpretation.

The harder but meaningful components of the model are those that require either or both of (a) nontrivial interpretation, or (b) judgment in real time about the appropriateness of showing the material. An example of this is an explanation that a student has a misconception about a concept. Such an explanation may well need to be synthesized just-in-time, shaded in such a way as to take into account the computed probabilities of the student’s holding various alternative beliefs about the concept. Making a judgment about whether to provide such an explanation might take into account both the strength of diagnosis achieved and what is known about the student’s willingness to accept such a report.

Material that is inappropriate for the student to see, if any, might include any of the following: (a) model variables for which no satisfactory explanation mechanism has yet been implemented in the system, (b) elements intended only for the system, such as checksums or other details, and (c) negative or even derogatory comments from a system or an exasperated teacher.

The dimension that affords the greatest challenge is probably the interpretive dimension. It is important because students cannot make use of complex assessment information if it is not presented in ways that are meaningful for them. Interpretation may be increasingly difficult as the sophistication of the assessment methods grows.

We can hope that the validation dimension might turn out to be just a technical implementation issue, and that once solved, it need not be something that students, teachers, and designers need to keep thinking about. However, mechanisms to engender trust in the validity of documents typically require the existence of trusted agents and trusted signatures, and there is always the potential for trouble there.

7. Acknowledgements

Thanks to Adam Carlson, Earl Hunt, Daryl Lawton, William Winn, the student developers of INFACT, the anonymous reviewers, and the National Science Foundation.

References

[1] P. Brna, J. Self, S. Bull, and H. Pain. Negotiated collaborative assessment through collaborative student modeling. In R. Morales, H. Pain, S. Bull and J. Kay (eds), *Proc. Workshop on Open, Interactive and Other Overt Approaches to Learner Modelling*, held in conjunction with AIED 99, Lemans, France, 1999, 35-42.

[2] S. Bull. Supporting learning with open learner models. *Proc. 4th Hellenic Conference with International Participation: Information and Communication Technologies in Education*, Athens, 2004.

[3] S. Bull, and H. Pain. 'Did I say what I think I said, and do you agree with me?': Inspecting and questioning the student model. In J. Greer (ed), *Proceedings of World Conference on Artificial Intelligence and Education*, AACE, Charlottesville VA, pp.501-508. 1995.

[4] S. Bull, A. T. McEvoy, and E. Reid. Learner models to promote reflection in combined desktop PC / mobile intelligent learning environments. *Proceedings of the Workshop on Learner Modelling for Reflection*, held in conjunction with AIED2003, Sydney, 2003.

[5] A. Corbett, and A. Bhatnagar. Student modeling in the ACT programming tutor: Adjusting a procedural learning model with declarative knowledge. In A. Jameson, C. Paris, and C. Tasso (eds), *User Modeling: Proceedings of the Sixth International Conference*. New York: Springer, 1997, pp.243-254.

[6] C. Hansen and G McCalla. Active open learner modelling. *Proceedings of the Workshop on Learner Modelling for Reflection*, held in conjunction with AIED2003, Sydney, 2003.

[7] J. Kay. A scrutable user modelling shell for user-adapted interaction. Ph.D. dissertation, Dept. of Computer Science, University of Sydney, Australia, 1998.

[8] J. Minstrell. Facets of students' knowledge and relevant instruction. In Duit, R., Goldberg, F., and Niedderer, H. (eds.), *Research in Physics Learning: Theoretical Issues and Empirical Studies*. Kiel, Germany: Kiel University, Institute for Science Education, 1992.

[9] J. Schoening. Creating lifelong learners through quality assurance. *Educational Technology and Society*, Vol. 5, No. 1, 2002. ISSN 1436-4522.

[10] S. Tanimoto. Beyond the naivety of grades: Educational record keeping for the twenty-first century. Technical Report 92-07-09, Dept. of Computer Science and Engineering, Univ. of Washington, Seattle WA, USA, July, 1992.

[11] S. Tanimoto. Distributed transcripts for online learning. *Journal of Interactive Media in Education,* 2001 (2) [www-jime.open.ac.uk/2001/2]. Published 10 Sept., 2001. ISSN:1365-893X.

[12] S. Tanimoto. Transparent interfaces: Design issues. In *Proceedings of the Workshop on Invisible and Transparent Interfaces, ITI 2004*, Gallipoli, Italy, 2004.

[13] S. Tanimoto, A. Carlson, J. Husted, E. Hunt, J. Larsson, D. Madigan, and J. Minstrell. Text forum features for small group discussions with facet-based pedagogy. *Proceedings of CSCL'02*, Boulder, CO., 2002.

[14] J. Vassileva, J. Greer, G. McCalla. Openness and disclosure in multi-agent learner models. In R. Morales, H. Pain, S. Bull and J. Kay (eds), *Proceedings of the Workshop on Open, Interactive, and Other Overt Approaches to Learner Modelling*, held in conjunction with AIED 99, Lemans, France, 1999.

[15] J.-D. Zapata-Rivera, J. Greer. Inspectable Bayesian student modelling servers in multi-agent tutoring systems. *International Journal of Human-Computer Studies*, Vol. 61, 2004, pp.535-563.