Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computers in Human Behavior 22 (2006) 131–148
Computers in Human Behavior
www.elsevier.com/locate/comphumbeh

Evaluating authoring tools for teachers as instructional designers

Shaaron Ainsworth *, Piers Fleming

School of Psychology and Learning Sciences Research Institute, ESRC Centre for Research in Development, Instruction and Training, University of Nottingham, University Park, NG7 2RD Nottingham, UK

Available online 2 March 2005

Abstract

The REDEEM authoring environment was developed to allow educators with no programming knowledge to design learning environments (simple Intelligent Tutoring Systems) for their students in a time-effective manner. The success of this approach depends on two key factors. Firstly, on the extent to which the authoring tool is usable by its intended author population (classroom teachers, university lecturers, adult trainers), and secondly, whether the resulting systems are effective at supporting learning. In this paper, a five year program is reviewed that evaluated the extent to which REDEEM has met these goals. The conclusion of the research is that in many ways REDEEM has exceeded the initial expectations for it, but that improvements to its design could further enhance its functionality.

Keywords: E-learning; Instructional design; Authoring tools; Classroom practice; Evaluation

1. Introduction

The Reusable Educational Design Environment and Engineering Methodology (REDEEM) authoring environment allows educators to create learning environments by importing computer-based training (CBT) as the domain content and

* Corresponding author. Tel.: +44 115 9515315; fax: +44 115 9515324.
E-mail address: sea@psychology.nottingham.ac.uk (S. Ainsworth).

0747-5632/$ - see front matter
doi:10.1016/j.chb.2005.01.010

then overlaying their views about how such material should be taught to their students. Accordingly, the authors' tasks are to select relevant material for their students and then make pedagogical decisions about such factors as the complexity of the topics, appropriate sequences through material, how to enhance interactivity and what teaching strategies would be most appropriate for particular types of learners.

Researchers coming from a strong Artificial Intelligence tradition might argue that the REDEEM approach is likely to prove ineffective. REDEEM learning environments are very simple systems and a number of studies have shown that increasing the 'intelligence' of systems improves learning outcomes. For example, Mark and Greer (1995) compared versions of a video recorder tutor. The 'smartest' one used sophisticated student modeling techniques to monitor learners' performance and could give detailed feedback on misconceptions, whereas a 'dumb' version allowed learners only one way to perform a task and provided only simple prompting. The 'smart' system decreased the number of steps, errors and time required for students to complete the post-test. Designers of intelligent tutoring systems (ITSs) hope that one day their systems will perform as well as expert human tutors, which, in itself, is a very high goal. Bloom (1984) found that one-to-one tutoring by expert tutors, when compared to traditional whole class teaching, improves students' learning by a 2 sigma effect size. This was the only pedagogical technique which had such a marked effect. Currently, state-of-the-art in ITSs is around a 1 sigma effect with evaluations of ITSs revealing effect sizes of between .4 and 1.2 compared to classroom teaching (e.g., Graesser, Person, Harter, & The Tutoring Research Group, 2001; Koedinger, Anderson, Hadley, & Mark, 1997).

However, the time and expertise needed to produce such 'clever' systems has meant that such ITSs have not yet achieved widespread application in schools, colleges or workplaces – creating an ITS is estimated to take between 300 and 1000 hours to produce an hour of instructional material (e.g., Murray, 1999). Moreover, the teams needed to create ITSs typically include knowledge engineers, software developers, domain experts, teaching experts, usability engineers, etc. Over the last 20 years, a number of authoring tools have been developed to address this problem (13 different systems are described by their creators in Murray, Blessing, & Ainsworth, 2003). These have been shown to substantially reduce the time needed, sometimes by an order of magnitude, and many authoring tools supplement the need for some members of the development team (see Murray, 2003 for an overview). However, research suggests that the difficulty of creating ITSs may not be the only reason that such systems are not found in every classroom – teachers may resist the introduction of such systems. For example, Major (1995) found that teachers were worried that ITSs would not reflect their own pedagogic concerns and approaches but would instead embody the beliefs of the system designers. One obvious solution is therefore to empower the teachers to be the designers of the environments that their students learn with in the classroom. But, most teachers are not trained as computer programmers and have only a very limited amount of time available for lesson preparation. Furthermore, it remains an empirical question whether teachers have appropriate expertise to create such systems even if tools are provided that are usable by non-programmers in a time-effective way.

Consequently, the value of REDEEM's approach is not easy to predict. On the positive side, non-programmers can use REDEEM to create courses in a very broad range of areas, and will be able to do so quicker with REDEEM than with than almost any other authoring environment. The people making the decisions about what and how students should learn will be professionals who make these decisions (in non-computational situations) everyday of their working lives. On the negative side, the systems created will have very shallow models of what they are teaching (relative to ITSs) and will be limited by the content of the imported material. REDEEM emphasizes macro-adaptation (selecting an environment for a particular student) over micro-adaptation (a response to particular action such as selecting the next action be it a new question or type of hint). Consequently, to evaluate the value of REDEEM's position on the tool usability versus ITS flexibility trade-off, it is necessary to explore the experiences of both the authors and the learners. This article briefly describes how REDEEM works and then presents the results of the evaluations.

2. System description

The REDEEM suite was developed in Click2Learn ToolBook Instructor and runs on Windows 95+. It consists of two main pieces of software (authoring tools and ITS shell) through which users interact with external courseware catalogues (see Fig. 1).

2.1. Courseware catalogues

Domain material in REDEEM is based on the idea of a courseware catalogue. It consists of pages of either Click2Learn ToolBook or HTML/Gifs/JPEGS. The ideal courseware for REDEEM presents discrete pages of material showing different

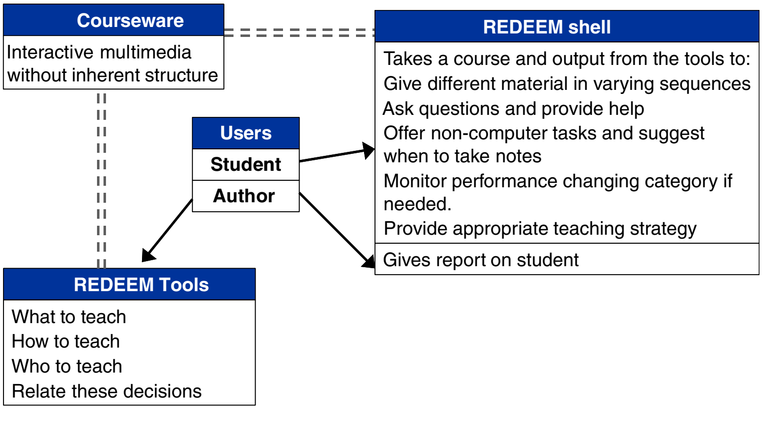

Fig. 1. REDEEM schematic.

aspects of the domain at varying levels of difficulty. Pages can contain multi-media, simulations, animations, questions and exercises. REDEEM does not model learners' actions on these objects.

2.2. Authoring tools

REDEEM's authoring tools decompose the teaching process into a number of separate components. Essentially, authors are asked to describe what they are teaching, whom they are teaching and how they would like to teach these students. This information is then combined by assigning particular teaching strategies and types of material to different learner groups.

2.2.1. What to teach

The first task that the author must perform is to give each page a name; other tasks can then be performed in any order. Sections are created by combining pages, which need not consist of contiguous pages in the underlying CBT. Pages can be in multiple sections. Sections are then described upon a number of dimensional ratings, (i.e., familiar, easy, general or introductory). Authors can describe relations between sections – for example, the prerequisite relation, which ensures that a section is not taught until prerequisite sections have been completed. Pages themselves are described in terms of these dimensional ratings and relations. These tools provide information that the system uses to make decisions about how to sequence material by calculating a weighted array of preferences (i.e., at any point it calculates which page would be its next best choice given the teachers' descriptions of pages and sections and its assumptions about teaching easy pages before difficult ones, familiar before unfamiliar, etc.).

The next stage is to add interactivity. Authors create questions (multiple choice, fill in the blank, multiple true, true–false or matching questions) and provide feedback that will explain to the student why an answer is correct. The author can create up to five different hints for each question, which ideally increase in specificity. Authors describe a number of characteristics of the question that the ITS shell uses to implement a specific teaching strategy (e.g., difficulty, pre or post-test). Authors can also associate a reflection point (which means that students are prompted to take notes) or non-computer task (which directs student attention to another activity) with a page.

2.2.2. Who to teach

The authors define a set of student categories at any degree of granularity, ranging from a whole class to an individual child. Teachers in these studies have tended to use performance-based measures (e.g., high flyer, struggler) or task-based measures (e.g., revising) or have combined these (e.g., high reviser). But, it is possible to use any dimension that authors find appropriate. The validity of performance-based categories can be evaluated against students' question performance. The shell can automatically change the category as the overall standard of the student (as defined in the shell's student model) changes. If this occurs then both content and teaching strategy may change as the system macro-adapts to the new category.

2.2.3. How to teach

REDEEM supports multiple teaching strategies. Different instructional principles can be embodied in various strategies by manipulating 'sliders'. Each slider in Fig. 2 has three discrete positions that result in different instruction. For example, teachers could create a 'Free Discovery' strategy where students choose the material they see, the questions they answer and the number of attempts allowed for each question, when to receive help and whether to perform non-computer-based tasks. In contrast, a teacher might create a 'Guided' strategy where students have no choice over material, when questions (of specific types and difficulties) are included and asked immediately after the relevant material has been presented and help is given on error with a limited number of attempts for each question. REDEEM can offer nearly 10 000 different teaching strategies each (very) subtly different to each other, although to date no author has created more than eight for a particular class of students.

2.2.4. Adapting material and strategies to student categories

By default, learners see all the material, but the author can choose to remove sections for a particular category (e.g., to focus on introductory material for learners who need more help or include extension activities for students who could benefit from more challenging tasks). Each student category is also given a teaching strategy. To date, authors have varied from creating a single preferred strategy to creating a unique strategy for each group or even for an individual student.

2.3. ITS shell

The REDEEM ITS shell uses the output of the authoring tools, together with its own default teaching knowledge, to interpret the courseware in such a way as to deliver adaptive, interactive instruction. The main role for the ITS shell is to deliver the course material to each student in the way that the teacher specified with the authoring tools. Tutorial actions available to the shell (depending upon the teaching strategy) are: teach new material; offer a question (and help if appropriate); suggest that students make notes on the on-line tool; offer a non-computer based task and by means of password protection check that is has been completed; or summarize students' progress. To achieve these functions, REDEEM employs a basic overlay model that records the system's understanding of the students' knowledge of an area. The values of the model change over the course of a session as the student sees new material and answers questions. This model is primarily used when student categories are performance based and determines if learners should change student category. The shell also maintains a student history. This is used to offer reports to the author either on an individual student's progress, a student category's progress or to give a report on the course.

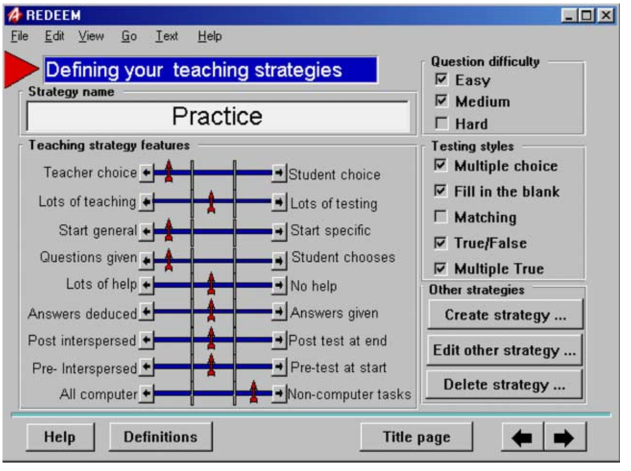

Fig. 2. Creating a teaching strategy.

3. Evaluating REDEEM

The success of REDEEM's approach will depend on two main factors – the usability of the authoring tools and the effectiveness of the created learning environments.

Firstly, teachers must be able to use REDEEM to create learning environments that match their requirements in a time-effective manner. REDEEM must be easy to learn and require only a short training period. Interfaces should be matched to authors' needs, providing simple to use tools and providing appropriate feedback on the consequences of their authoring decisions. REDEEM, because its intended users are teaching practitioners, requires higher usability standards than is necessary in authoring tools aimed at more specialized authors. Authors should be able to create environments that reflect their own pedagogic principles and that meet the learning needs of their learners so REDEEM must offer appropriate teaching decisions in way that match teacher's ontologization of the teaching-learning process. Given the time-constrained nature of the process, it is equally important to limit the functionality that authors do not require. For example, if all authors make very similar decisions about a teaching strategy dimension, then choice in this dimension is redundant and should become hard-coded into future systems.

Secondly, the environments that teachers create should be effective at supporting their students' learning. Ideally, learners will come to understand the subject matter more completely, will have found the experience of learning motivating and should reach the desired outcome in a more time efficient way. However, to prove that a system led to successful outcomes is not an easy task. To be done successfully, large-scale experiments are often needed. Furthermore, the effectiveness of any learning environment is influenced by its context of use. For REDEEM, the evaluation problem is compounded by the fact that a learning environment depends on the author decisions, the choices the authoring tools offered, the shell's interpretation and delivery of these decisions and externally created domain material. The nature of the control in such evaluations is very important. Often, a system may be compared against a human teacher or learning through textbooks. But, such comparisons have only limited value in many cases because the differences between the two are so large that little sensible can be said about why any difference in learning processes or outcomes occurred. One alternative often used in Artificial Intelligence is an ablation experiment where particular design features are removed and performance of the systems compared (e.g., Cohen & Howe, 1988). They also allow analysis of the contribution that specific system features bring to the learning experience. In REDEEM's case this seems a very appropriate methodology, as a question that should be answered is whether investing teachers' time in REDEEMing existing CBT is worthwhile. Hence, a number of experiments have been conducted that compared learning with the original CBT to learning with the REDEEMed CBT.

4. Usability of the authoring tools

The REDEEM authoring tools have now been used by many different types of author: including primary (K-12) and secondary (high) school teachers, lecturers, researchers, military instructors, and student teachers. This has provided an unprecedented opportunity to explore their usability and functionality over many user groups. Overall, it was found that it is possible for all these classes of user to use the tools to express, represent and assess their teaching knowledge to create an ITS within a feasible time scale. Initial training in the use of the REDEEM tools requires between 1 and 2 h. No author has found the overall decomposition of teaching process incompatible with his or her approach. REDEEM's reliance on graphical manipulation of sliders and form-fill style interview tools has proved simple and easy for authors to use (see below for the single exception).

To stand a realistic chance of use in the classroom, authoring tools must also be efficient of teachers' time. In one study (Ainsworth, Grimshaw, & Underwood, 1999) authors took between 6 and 11 h (to author a 4 h course on 'Understanding Shapes') – a ratio of around 3 h per hour of instruction. Ainsworth and Grimshaw (2002) found that a teacher took less than 25 h to create two environments (around 8 h of instruction). Navy authors began by requiring 10 h per chapter (around 6:1), which dropped to 6 h by the end of authoring (around 3:1) (Ainsworth, Williams, & Wood, 2001). Furthermore, the time consuming aspects of the ITS development lies in the domain authoring. When trainee teachers were presented with a previously authored course that they just had to individualize to their students, they only required 90 min to customize the 4 h course (Hayes, Underwood, Ainsworth, & Grimshaw, 2001).

For the most part authors wanted the functionality that REDEEM provides and they could use it to create learning environments that reflect their own pedagogic preferences. However, the focus of authors' attention was radically different depending upon whether they were classroom teachers or military instructors. The teachers wanted to structure the domain material in ways that reflected their own beliefs (e.g., by creating sections of differing complexity they could differentiate course material for different groups of learners). Classroom teachers also used REDEEM to differentiate their teaching strategies. Five out of six teachers created at least five student categories and associated a unique teaching strategy with each one. In contrast, the military authors were much less interested in creating individualized experiences. They used REDEEM to create a single teaching strategy that reflected their own preferred strategy rather than the courseware designers' strategy. What both groups had in common was the way they increased interactivity by adding questions and reflection points. Adding this interactivity was one of the time-consuming tasks, not because of its complexity but for the most part because the authors valued it so highly.

After authoring, the designs of the systems were recognized by all the authors as closely related to their instructional approach. They expressed more satisfaction with their learning environment than with the original CBT and felt they would be more likely to accept their personal REDEEM system in their classroom than the CBT alone or another author's learning environment.

However, these studies have revealed some interesting dilemmas about REDEEM's design. Unsurprisingly, a number of small changes to the REDEEM interface have been made over the years based on author feedback (e.g., more undo functions), but more interesting design issues have also been revealed.

Firstly, in common with most ITS authoring tools, REDEEM requires users to shift from story boarding to knowledge-based authoring (e.g., Major, 1995). For example, authors describe characteristics of the page (e.g., familiarity and complexity) and then the shell computes a route. However, this proved too time-consuming, was unpopular and thus was resisted. Knowledge-based authoring ignores the important role that narrative plays for authors and for learners when they are interacting with new material. This experience suggests that this call for knowledge-based authoring may have been overstated, at least for designing declarative instruction.

A second related dilemma is whether teachers should be encouraged to envisage a future where they work in collaboration with more intelligent software or instead provide them with tools to create software to fulfill their needs today. Compared to the functionality that REDEEM offers, it is apparent that teachers mostly used REDEEM to create customized CBT rather than adaptive ITSs. For example, authors often wanted control over the order of presentation of material, like CBT, and felt uncomfortable releasing this role to REDEEM. No teacher has chosen to let the system macro-adapt based on student performance.

Thirdly, it is apparent that more attention should be paid to how the REDEEM tools support authors in understanding the consequences of their decisions. In the first version, the only way to check the consequences of an authoring decision was to swap to the ITS shell and behave as a student. This is easily achieved and is an excellent way to check the consequences of adjusting features such as teaching strategies, but it is a poor way of checking micro-structural issues such as position of questions. Consequently, visualization tools have been developed so that authors can more easily see the consequences of their actions. However, as any visualization will remain an abstraction from what might actually occur in the ITS shell (given that an ITS responds to a student's performance) this raises a host of other design problems. For example, graphical representations are not an ideal way to represent information with a lot of uncertainty (e.g., Stenning & Oberlander, 1995).

Finally, time is crucial. Authors need to create learning environments in a time-efficient way. REDEEM achieves a ratio of around 4 h of authoring to 1 h of instruction when a trained author, familiar with the domain and with teaching, creates an ITS from imported domain material. This ratio substantially undercuts the majority of ITS authoring tools, even allowing for the time involved in finding appropriate domain material. A lesson from these studies was 'this is still not fast enough'. One tension here was the need for control and ownership of the software versus the time taken to customize it. As discussed above, authors were uncomfortable allowing the computer to determine sequence and so spent much time ensuring the system was hard-coded to their preferred sequence. They often spent a considerable amount of time on details that potentially may not be too important (e.g., one teacher edited another ones help messages to include a full stop after each message). It would be interesting to see how authors would behave if they were routinely authoring many courses. However, on the positive side one of the features that teachers appreciated most was the way that REDEEM allows quick assignment of different strategies and content to different student categories. Analysis of authoring times suggests that this is normally achieved in around 30 min, irrespective of the size of the course.

5. Effectiveness of the REDEEM learning environments

Analysis of the usability and functionality of REDEEM has mostly revealed positive results. However, for REDEEM to be truly useful, it should also make learning more effective. Five studies have been conducted with REDEEM with a variety of authors (classroom teachers, military trainers and researchers) and learners (school children, university students, military personnel). Three different types of content material have also been used: Genetics, which was developed for REDEEM; Communications and Information Systems Protocols (CISP), which was developed for use in Naval classrooms as standalone CBT; and PC and Networking, which was adapted from legacy CBT in use in the Navy. All of the studies have taken the same basic approach. As REDEEM is based on non-intelligent CBT, learning outcomes from those students working with CBT can be compared to learning outcomes with REDEEM learning environments. If learning outcomes are higher with REDEEM, then the conclusion that the REDEEM/Author partnership in the situation provided better support for learning than the non-intelligent courseware is warranted. Here, the details of one of these studies are presented before summarizing the results from all the studies.

Table 1
Comparison of REDEEM and CBT

	REDEEM features
Content	REDEEM and CBT provide the same content but REDEEM provides section introductions and summarises progress
Level of control	Author controlled, so similar to CBT
Questions	20 questions per course. Questions asked at the end of a section. Students allowed multiple attempts to answer question correctly. None of these questions are in CBT
Help	On request and error with between 1 and 5 hints per question
Reflection points	11 per course in REDEEM. General advice to make notes in CBT

5.1. The PC and Networking student study

The courseware used in this study was based on two courses developed by the Royal Naval School of Education and Training Technology for use in Naval colleges. The first course provided an introduction to personal computing (PC) and was 60 pages long and the second course introduced working with networks and was 52 pages long (Net). The courses consist of text and graphic declarative material with some multimedia, animation and simple exercises. Navigation through the course is linear. They were adapted for use in this study by removing Navy specific material and by extending the courses to include some more difficult concepts.

A researcher then used the REDEEM Tools to create two simple REDEEM ITSs (adapting the authoring that had been performed originally by a Naval trainer). The REDEEM differentiation features were not used, as the researchers had no personal knowledge of the students in the classes. This resulted in one learner category; so all students saw the same material and received the same teaching strategy. Thus, the main difference between the two courses is in the interactivity supported by REDEEM rather than in the use of individualization. As such both the ITSs and the experimental design are considerably simplified compared to some of the other studies (see Table 4). The main authoring decisions and, in consequence, the additional features in the REDEEM courses are summarized in Table 1.

5.1.1. Participants

Participants were 33 University of Nottingham undergraduates, ranging in age from 19 to 24 years. Students studying any subjects other than the Computer Sciences were accepted. Participants were paid £20.

5.1.2. Material

A 60 item multiple-choice test was developed (one correct answer and three distracters). It consisted of 30 questions on the Personal Computing and 30 on Networking.

5.1.3. Design

A crossover design was used such that all participants received one course under REDEEM and one as CBT only, that is, half received REDEEM Personal Computing and CBT Networking and half CBT PC and REDEEM Networking.

5.1.4. Procedure

The experiment was carried out over four 2 h sessions on separate days, in a large computer lab at the University of Nottingham. Each session involved one of the two groups studying one course. The experimental procedure was as follows:

1. Participants took the Personal Computing test (time was unlimited) and then were randomly assigned to study either the REDEEM or CBT PC Course.
2. A brief explanation was given of the basic purpose of the study and the application they were about to use. For the REDEEM courses, a quick reference guide on using the system was made available. Two experimenters were on hand throughout the procedure to help with any questions or difficulties relating to the use of the software.
3. Participants were allowed as much time as they needed to work through the material. On completion they informed the experimenters, who gave participants the post-test measure to complete.
4. Each participant took part in two of the sessions and the same procedure was repeated for each of the remaining sessions.

5.2. Results

To examine the effects of the intervention, a [2 by 2 by 2] mixed design ANOVA was carried out on the pre-test and post-test data. The design of the analysis was 2(PC, Net) by 2(pre-test, post-test) with a between subjects factor of order of environment (REDEEM PC/CBT Net, REDEEM Net/CBT PC). Two participants (one from each group) were excluded for non-completion of the procedure (see Table 2).

Analysis revealed a significant main effect of time ($F(1,31) = 373.84$, MSE = 38.86, $p < .0001$), with subjects scoring higher at post-test. This was modified by significant interaction between course and time ($F(1,31) = 5.23$, MSE = 38.86,

Table 2
Pre and post-test percentage scores by course and environment

	REDEEM				CBT			
	PC (n = 17)		Network (n = 16)		PC (n = 16)		Network (n = 17)	
	Mean (%)	SD	Mean (%)	SD	Mean (%)	SD	Mean (%)	SD
Pre-test	35.83	9.62	35.70	9.04	40.59	9.59	33.84	7.98
Post-test	89.38	9.62	92.90	7.56	84.31	9.56	83.84	9.22

$p < .05$) and a significant three way interaction between time, course and environment ($F(1,31) = 29.81$, MSE = 38.86, $p < .0001$). To simplify the analysis and because there were no significant differences between the conditions at pre-test, improvements scores per subject were calculated (post-test–pre-test) and analyzed using a [2 by 2] mixed ANOVA with condition and course as factors (Fig. 3). This showed a significant effect of course ($F(1,31) = 5.23$, MSE = 38.86, $p < .05$) where scores for Networking increased more than scores for PC. It also revealed a significant interaction ($F(1,31) = 15.37$, MSE = 77.71, $p < .001$). Simple main effects analysis confirmed that subjects improved their scores more on the course they experienced through REDEEM ($F(1,62) = 7.82$, MSE = 101.61, $p < .005$ and ($F(1,62) = 4.21$, MSE = 101.61, $p < .05$) for PC and Networking, respectively). However, the difference between REDEEM and CBT was only significant for the CBT first condition $F(1,31) = 19.8$, MSE = 77.8, ($p < .001$), that is those students transferring from CBT to REDEEM improved their performance on the REDEEM course, but those students transferring from REDEEM to CBT did not decrease their performance. A similar analysis controlling for time on task revealed an identical pattern of results.

Two types of process measure were calculated for both REDEEM and CBT – the number of notes written and the amount of time spent learning. To calculate the notes written, both words written on paper and in REDEEM were totalled. Analysis by [2 by 2] mixed ANOVA with factors of condition and course showed a significant effect of condition – more notes were written by the participants who studied with REDEEM first $F(1,31) = 9.06$, MSE = 34301, ($p < .005$) (see Fig. 4). Number of notes written did not correlate with any measure of performance (pre-test, post-test, improvement) but does with time spent learning ($r = .87$, $p < .001$).

Analysis of the time data showed that students spent considerably longer on the PC course ($F(1,31) = 20.4$, MSE = 17052, $p < .001$) and that those students in the REDEEM first condition spent longer learning overall ($F(1,31) = 9.451$, MSE = 34989, $p < .005$) (see Fig. 5). Time spent learning with either environment

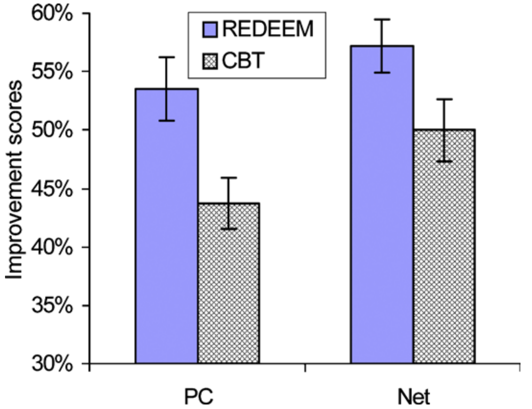

Fig. 3. Improvement scores by course and environment.

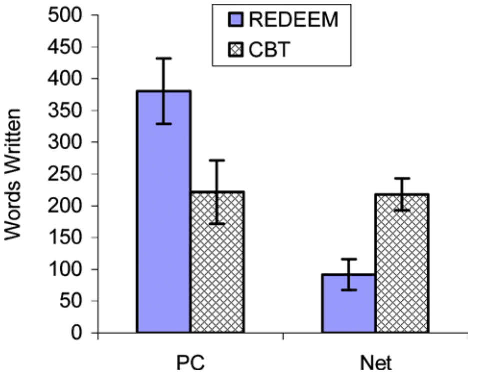

Fig. 4. Notes written by course and environment.

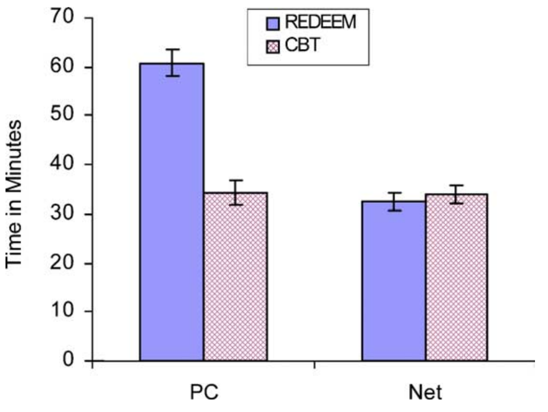

Fig. 5. Time spent learning by course and environment.

does not correlate with any measure of performance (pre-test, post-test, improvement).

REDEEM logs information about how many attempts a student requires to answer the question correctly and the number of hints either provided or requested. These measures were analyzed to examine if there was any systematic relationship between behavior with REDEEM, students' performance, and prior knowledge. Data is presented for the learner's REDEEM interaction, irrespective of whether it was the PC or Networking course (see Table 3).

Overall students who knew more at pre-test learnt less during the intervention. This is due to a ceiling effect as overall post-test scores were nearly 90%. Improvement scores do not correlate with total hints, or attempts at question. However, if the analysis is repeated with pre-test scores partialled out then those students who took more attempts to answer the question were found to have learnt the least ($r = -50$, $p < .01$).

Table 3
Correlation between pre-test, improvement scores, hints, and attempts at the question

	1	2	3	4
1. Pre-test score	–	−.70***	−.33	.52**
2. Improvement score	–	–	.07	−.12
3. Total hints	–	–	–	−50**
4. Attempts at question	–	–	–	–

Note: *$p < .5$, **$p < .01$, ***$p < .01$ (two tailed test of significance).

6. General discussion

This evaluation showed that REDEEM could be used to create learning environments that are more effective than the CBT they are based on. This study can be placed in context with four other studies which have used a similar design to address this question in a variety of different contexts (Table 4). On average, REDEEM has led to 30% improvement from pre-test to post-test, whereas CBT increased scores by 23%. This advantage for REDEEM ITSs translates into an average effect size of .51. This compares well to non-expert human individual tutors (an average of .4 sigma, Cohen, Kulik, & Kulik, 1982) and is around .5 below the best performing full-blown ITSs.

There are a number of things to observe about the results. Firstly, the variability in the outcomes is incredible and is in fact even greater for individual courses (the highest effect size is for REDEEM Genetics 1 versus CBT Genetics 1 in study 2 where the effect size was 1.33 sigmas). Interpretation of these results requires overcoming a large credit assignment problem. There is no consistent relationship between whether the course was authored by a researcher or a practitioner, the topic taught, whether the study was conducted in an artificial situation or in a realistic context, and whether REDEEM's macro-adaptive features were used to create learning environments for specific learner categories.

However, analysis of process data does go some way to explain these results. Many studies showed significant correlations between amount of notes written, and time spent learning with REDEEM and learning outcomes. All studies found a significant relationship between percentage of questions answered correctly first time during the learning session and outcome, (often when prior knowledge is partialled out). Unsurprisingly, learners who took advantages of REDEEM's features learnt more than those who did not. The two studies without a REDEEM benefit were also the ones with the highest degree of students who were least interested in learning this material. Analysis of the process logs suggests a wide variation in how much learners interacted with the systems. Some studies have also presented learners with questionnaires asking them to identify what they felt most benefited their understanding. The most consistent response has been questions with appropriate feedback and hints are viewed as most beneficial. It remains a valuable lesson to system designers and researchers – adding features to enhance learning can only impact when learners choose to engage with them.

Table 4
Summary of learning outcomes studies conducted with REDEEM

Study	Subjects	Course	Time	Location	Author	Versions	Gain (%)	Effect size
1. Genetics at Uni.	86 14–16 year	Genetics	8 h	University	Teacher	5 varying content and strategies	RED = 10 CBT = 8	.21
2. Genetics in school	15 14–16 year	Genetics	8 h	School	Teacher	3 varying content	RED = 16 CBT = 8	.82*
3. Navy	19 17–22 year	CISP	15 h	HMS Collingwood	Trainer	1	RED = 21 CBT = 22	−.04
4. Student	25 20–28 year	PC and networks	3 h	University	Trainer/researcher	1	RED = 53 CBT = 44	.82*
5. RAF	16 20–45 year	PC and networks	3 h	RAF Waddington	Trainer/researcher	1	RED = 47 CBT = 32	.76*

Note: * = statistically significant difference.
Effect size = (gain in experimental condition – gain in control)/(standard deviation in the gain of the control group).

REDEEM differs from the underlying CBT in a number of ways (depending upon authors' decisions), but the two most crucial are the interactive and the macro-adaptation features. The results of the studies suggest that any advantage of REDEEM was due more to increasing interactivity than to macro-adaptation. Studies 1 and 2 included macro-adaptation to different learner categories (either by content or by strategy and content) but were no more successful than studies with only a single student category. That degree of interaction predicted learning outcomes does not seem contentious, but the question that remains is why benefits from macro-adaptation were not observed. The teachers use of REDEEM's content and teaching strategy adaptation features are primarily in line with that of the research literature (see Ainsworth & Grimshaw, 2002). However, the two teachers who used macro-adaptation categorized learners solely in terms of perceived ability, but many other variables can be explored (e.g., learning style, working memory capacity, self-regulatory skills, visualizer/verbalizer, gender, high anxiety/low anxiety, level of cognitive development) and potentially these should have been included. Furthermore, many of these factors may show up in laboratory studies, but their effect size may be somewhat weak and they may have little impact in the classroom. The number of students in each category would often not have been sufficient to allow identification of benefits unless they were very substantial. Moreover, from this design they might be difficult to identify. For example, if an author assigned a unique teaching strategy to every category of learner and they all made equal gains, does this mean that the strategies were ideally targeted or that they had no effect? Consequently, further research is needed to examine the educational significance of macro-adaptation and to consider which are the most important learner characteristics and strategy dimensions. It remains an open question whether it is educationally effective to perform pre-test or psychometric tests to assign students to different learning environments.

Finally, almost all the studies found a greater advantage for learners who received REDEEM first rather than CBT first. The direction of this benefit is positive, that is students transferring from REDEEM to CBT do better than predicted. This may suggest that learning with an environment that encourages greater interactivity may help when learning with an environment without these beneficial features. It is also worth noting that this will have served to reduce any differences between learning outcomes in the experiments.

7. Conclusion

The REDEEM authoring environment was developed as a response to a desire for teachers and trainers to be involved in the development of learning environments for their classroom. The evaluation studies that have been conducted over the last five years suggest teachers can be successful instructional designers if given usable tools that offer appropriate functionality in a time-efficient way. However, even simple tools still require educators to engage in complex inferencing about the consequences of their authoring decisions. The learning experiences of their students has generally been enhanced, especially when learners take advantage of the interactive features authored by the teachers. The studies have revealed that teachers have complex models of teaching and learning which underpin their work. They do not adhere to simple one-dimensional accounts of learning. The flexibility that REDEEM provides for adapting CBT should allow such CBT to be used in ways that are appropriate to different teacher's methods. It is often said that we do not expect every teacher to write their own textbook, why should we expect them to design their own ITS? However, all teachers do customize their textbooks to use in their classroom by suggesting an order to read chapters, explaining difficult terms, providing exercises and worksheets, etc. REDEEM supports the teaching and learning process in the same way – the domain material is not created by teachers but its pedagogical functionality is. Hence, although the REDEEM approach may not be suitable for all circumstances, much can be gained by providing teachers with usable authoring tools to create adapted learning environments.

Acknowledgments

This work was supported by the ESRC at the ESRC Centre for Research in Development, Instruction and Training and by the Office of Naval Research under Grant No. N000 14-99-1-0777. Many researchers participated in this project including Nigel Major, Shirley Grimshaw, Alan Richardson, Mary Hayes, Jean Underwood, Ben Williams and David Wood. Iona Bradley, Steve Bruce-Lomba, Sue Cavendish, Wendy Crossland, Ruth Guy-Clarke, Sue Hewes, CPO Pete Skyrme and CRS JJ O'Neil were invaluable as the authors.

References

Ainsworth, S. E., & Grimshaw, S. K. (2002). Are ITSs created with the REDEEM authoring tool more effective than 'dumb' courseware? In S. A. Cerri, G. Gouardéres, & F. Paraguaçu (Eds.), *Proceedings of the 6th international conference on intelligent tutoring systems* (pp. 883–892). Berlin: Springer-Verlag.

Ainsworth, S. E., Grimshaw, S., & Underwood, J. (1999). Teachers implementing pedagogy through REDEEM. *Computers and Education, 33*(2–3), 171–187.

Ainsworth, S. E., Williams, B., & Wood, D. (2001). Using the REDEEM ITS authoring environment in naval training. In *Proceedings of the IEEE international conference on advanced learning technologies* (pp. 189–192).

Bloom, B. S. (1984). The 2 sigma problem: the search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher, 13*(6), 4–16.

Cohen, P. R., & Howe, A. (1988). How evaluation guides AI research. *AI Magazine, 9*, 35–43.

Cohen, P. A., Kulik, J. A., & Kulik, C. C. (1982). Educational outcomes of tutoring: a metaanalysis of findings. *American Educational Research Journal, 19*, 237–248.

Graesser, A. C., Person, N. K., & Harter, D.the Tutoring Research Group. (2001). Teaching tactics and dialog in AutoTutor. *International Journal of Artificial Intelligence in Education, 12*, 257–279.

Hayes, M., Underwood, J. D., Ainsworth, S. E., & Grimshaw, S. K. (2001). Exploring pedagogy through a virtual class. Presented at *Computers and Learning 2001*, Warwick.

Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1997). Intelligent tutoring goes to school in the big city. *International Journal of AI in Education, 8*, 30–43.

Major, N. (1995). Modelling teaching strategies. *Journal of AI in Education, 6*(2), 117–152.

Mark, M., & Greer, J. E. (1995). The VCR tutor: effective instruction for device operation. *The Journal of the Learning Sciences, 4*, 209–246.

Murray, T. (1999). Authoring intelligent tutoring systems: an analysis of the state of the art. *International Journal of AI in Education, 10*, 98–129.

Murray, T. (2003). An overview of intelligent tutoring system authoring tools: updated analysis of the state of the art. In T. Murray, S. Blessing, & S. E. Ainsworth (Eds.), *Tools for advanced technology learning environments* (pp. 491–544). Amsterdam: Kluwer.

Murray, T., Blessing, S., & Ainsworth, S. E. (Eds.). (2003). *Tools for advanced technology learning environments*. Amsterdam: Kluwer.

Stenning, K., & Oberlander, J. (1995). A cognitive theory of graphical and linguistic reasoning: logic and implementation. *Cognitive Science*, 97–140.