# Computer Based Assessment (Volume 2): Case studies in Science and Computing

*Edited by Dan Charman and Andrew Elmes*

Science Education Enhancement and Development

SEED

# AN INTRODUCTION TO SEED

The programme for Science Education Enhancement and Development ('SEED') is based in the Faculty of Science at the University of Plymouth. It is resourced principally by the Higher Education Funding Council for England through its Fund for the Development of Teaching and Learning. Additional support has been received from many areas of the University and particularly from Academic and Information Services. SEED builds on the success of the University of Plymouth Science Faculty in the national Teaching Quality Assessment system where Plymouth achieved 'excellence' in Environmental Science, Geography, Geology and Oceanography.

SEED's overall aim is to develop, document and disseminate good practice in Science teaching and learning. The programme consists of a series of projects in areas such as lab-work, field-work, graduate teaching assistants and computer-aided learning, which are itemised inside the back cover. Most are based in the Science Faculty but some are being taken forward by staff in the University's School of Computing and in Educational Development Services (EDS). All the projects are linked to dissemination partners in other institutions who act as external advisors, ensure that SEED's outputs are capable of being used in other institutions and help to disseminate and embed SEED's end-products.

Anyone wanting further details on the SEED programme is welcome to contact Brian Chalkley or Andy Elmes at the address below. Contact details for the individual project leaders are available inside the back cover.

## The SEED Programme

Faculty of Science
University of Plymouth
Drake Circus
Plymouth PL4 8AA
Tel: 01752 233530 or 01752 233053
Fax: 01752 233534 or 01752 233054

Programme Manager: Brian Chalkley
e-mail: bchalkley@plymouth.ac.uk

Programme Officer: Andrew Elmes
e-mail: aelmes@plymouth.ac.uk
http://www.science.plym.ac.uk/departments/seed/

## Acknowledgements

The SEED Programme would like to thank Joanna Bull (University of Luton), Professor Harold Silver and numerous colleagues at the University of Plymouth for kindly reading and providing helpful comments on the draft of this handbook. Thanks also to Arnold McAlpin (CTI Computing, University of Ulster at Jordanstown) for his help in identifying case study material contacts and to all the authors of the case studies (Volume 2), who have provided some excellent examples of CBA in action. Finally, many thanks to Gary Batten, (Media Services, University of Plymouth) for his help with the design and layout of the handbook.

# Computer Based Assessment (Volume 2): Case studies in Science and Computing

**Edited by Dan Charman and Andrew Elmes**
Department of Geographical Sciences and
Science Education Enhancement and Development (SEED)
University of Plymouth



**ISBN 1-84102-02-7**

## Contents

Page

# Introduction: Rationale for and use of Volume 2

This is the second volume of a handbook on computer-based assessment (CBA) in Higher Education. The first volume ("A guide to good practice in computer based assessment") reviewed the rationale for, and the application of, CBA in order to provide general guidance on the design and implementation of CBA. However, some of the best practical advice on actual implementation comes from those who have attempted to find solutions to their own assessment requirements. This second volume provides this via a series of case studies written by teaching staff with relevant experience.

Until you try out CBA you won't really be sure if it is worthwhile. However, a lot can be learned from hearing about other people's experiences and we hope this volume will be useful for precisely that purpose. It is also worth finding people in your own department/institution who have some experience of CBA, especially if you think there may be logistical or practical limitations outside of your immediate control. We have provided contact names, addresses and emails of the people who have provided the case studies in this volume and recommend that you contact them directly if you want to know more about any of the examples. See also the other contact addresses in the further information section at the back of the handbook.

The case studies detailed in this volume are not exhaustive in approach or subject area but we hope they are a reasonable cross section of what is currently being developed and used in HE and derive from both 'old' and 'new' (post-1992) Universities in the UK . There is a bias towards science and computing in line with the original purpose of the SEED project (through which this handbook was resourced), but much of the advice and many of the lessons are equally applicable in other disciplines and we hope they will be useful to a variety of academics across Higher Education.

# CASE STUDY ONE:
# Computer-marked tests in geography and geology

Joe Angseeing

# Case Study One: Computer-marked tests in geography and geology


### *Author and contact:*

Joe Angseeing
Department of Geography and Geology
Cheltenham & Gloucester College of Higher Education
PO Box 220, The Park
Cheltenham
Gloucester
GL50 2QF
UK

### *Email:*

JoeA@chelt.ac.uk



### *Key-words:*

computer-based assessment, multiple-choice tests, objective tests, question analysis and facility.



### *Outline:*

First year students in geography and geology have been partly examined using computer-based assessments since 1985. Our current practice is to use multiple-choice tests which are undertaken at a computer terminal. Marking of test answers is objective, absolutely consistent and extremely rapid, and output includes an analysis of facility and of the distractors chosen.


### *Context:*

Cheltenham & Gloucester College operates a modular degree scheme based on a semester system, with 15 weeks per semester. First year modules have relatively large student numbers (up to 120 in the Geography and Geology Department) because they are taken by most minor, as well as major and joint, subject students. Computer-based tests are used in geology (up to 60 students), physical geography and geography modules, and have been used in modules in earth's resources. They help to ease the assessment bottleneck that occurs twice a year at the end of each semester.

Currently, automated testing is used in five of the Department's eleven first-year modules, an increase over previous years; between 1985-1994 only 10-25% of level I modules were assessed in this way. In two modules they are organised as part of the formal examination programme at the end of the semester, but in the other three they occur within the module as part of the continuous assessment (see Table 1).

### *Description*

All the computer-based tests are undertaken at a computer-terminal. The software used is called CMAPROG and was developed in house. It is loaded up in advance by the invigilator, who also distributes any other ancillary material. Students fill in a cover-sheet, noting their student number and terminal number, and begin. The software is self-timing, starting from the point where the question file is loaded.

The software is text-based - it presents the question on-screen and an array of possible answers. Examinees respond by typing in a key (usually a number in the range 1 to 6) for the selected response.

About half the tests in current use refer to ancillary material - mostly specimens in geology exams and printed graphs, diagrams and tables in geography. The method of selecting (or editing) an answer means that the time limit is not a problem. Candidates are set between 60-80 questions per hour and usually finish with 10-20 minutes to spare. Large modules may require two sittings, which means that members of the first test-group have to remain supervised until the second group is seated.

**Table 1. Computer-marked tests in geography and geology modules at CGCHE: assessment timing (within a 15-week semester), weighting and duration.**

| Module title | Assessment in week: | Computer test weighting | other test weighting | Test duration (minutes) |
|---|---|---|---|---|
| Earth Materials | 6 | 20% | 50% | 30 |
| Earth structure and evolution | 14 - 15 | 25% | 25% | 45 |
| Environmental data handling | 14 - 15 | 50% | - | 60 |
| Global Environmental Issues | 10 | 25% | 25% | 60 |
| Environmental Systems | i) 6<br>ii) 10 | i) 35%<br>ii) 35% | -<br>- | i) 60<br>ii) 60 |

It takes less than 5 seconds for a separate computer program to mark 120 answer files, although printing out results, including a basic question analysis, takes a few minutes. The question analysis lists (i) the percentage of candidates who were successful on each question and (ii) a table showing how often each distractor was chosen.

### *Resource Implications*

**Physical resources:** Sufficient terminals and specimen sets are required for each member of the module, or for half the group if two sittings are planned. Running an exam in two sittings is fairly straightforward, but more than two becomes a logistical problem. Generally there will be no resource problem as most institutions will have computer rooms for class teaching, though sometimes the spacing of the machines means that only alternate places can be used in an exam. The specimens required are no more than would be used for a conventional exam in subjects such as geology. If investment in teaching computers has already occurred then the only finance that might be required is to acquire a suitable software package (usually less than the cost of one computer).

**Staff time:** This can be broken down into three areas:

- **Preparation of the test**

  Preparation time is longer than for a conventional written paper on the first occasion that a module uses multiple choice testing, although converting existing paper-based tests to computer-format will typically take only 4-6 hours. A new 80-question test will take from 12-24 hours of work depending on staff experience and question complexity. It is advisable that all questions are proof-read to ensure that they are of appropriate difficulty and unambiguous, adding 4-6 hours to the process. Once a test bank has been established, about 4-6 hours of staff time will be sufficient to select an appropriate mix of questions. It is better to use a majority of questions of known facility if possible, so new questions are only required to keep pace with changes in the module curriculum.

- **Invigilation**

  Invigilation staffing is comparable to that for conventional exams. Invigilators have to type in the passwords that restrict access to the test software in lieu of dishing out blocks of paper.

- **Marking**

  Marking is rapid and painless. Overall, computer-based testing requires a greater investment of staff time than conventional examinations, but only for the first year or two. By the end of three years the staff time demands should have broken even; beyond that is net profit.

### *Student performance*

Very high marks are possible, as are low scores, so the coefficient of variation is higher than for conventional written exams. In one extreme case the first computer-based test we constructed for 'Environmental Data Handling' produced a bimodal distribution: 40% or so of the candidates were on top of the work and formed a mode at over 60%, while the rest of the candidates found the exam quite difficult and clumped about a mode at 40%; only 3 of nearly 60 examinees scored between 50-59%. This resulted from about 25% of the questions having a high difficulty but less than 10% being easy, this is where experienced question-setting or reviewing is vital to ensure that there is a range of questions with respect to facility.

Correlations in student performance between computer-based tests and other forms of assessment are quite high - there was a correlation coefficient of between 0.55 and 0.70 in three recent comparisons between computer-based and written answers in geology. The question analysis highlights areas of difficulty both in the course and also general processes such as the manipulation of units and the sequences of reasoning in making inferences.

### *Evaluation*

The computer-based tests used at Cheltenham offer similar advantages and disadvantages to paper-based multiple-choice tests including:

**Advantages:**

- Consistent marking
- Error-free totalling
- Easy question analysis
- No discrimination against candidates who write slowly
- Time-saving once a question bank has been established and tested
- They compel candidates to revise the whole of a course
- They can provide feedback to students when they are used within a module

**Weaknesses:**

- They do not test precise recall (e.g. spelling) of technical terms
- They do not assess synthesis
- English expression - both grammar and clarity - are not tested
- They do not assess the construction of an argument
- They do not assess systematic breakdown and treatment of broad questions

Overall, the advantages make these techniques worth employing, but the disadvantages limit their application. Examiners have most to gain with large first-year modules, and their use in this department is restricted to level I. They do assess more than recall, and can be used to test numerical manipulation, practical observation and the application of sequences of reasoning using known relationships. They cannot replace essay-type answers where examiners want to assess reasoning, argument and synthesis. In this department we have experimented with open-ended computer-marked tests (Angseesing, 1989) but have found that this approach requires much more marking time, even after several runs of a question have built up a substantial answer bank.

### *Key advice*

- Have a contingency plan for computer breakdowns (these are not common, but we have had both individual monitors and keyboards die during exams). The best arrangement is to use a software package that continuously (and automatically) saves work and allows it to be retrieved once the candidate's computer has been restarted.
- Allow plenty of time to set up the first question set and ensure that it is adequately proof read.
- Maybe have a specimen question set available to students in advance, so they can familiarise themselves with the question style and how the software runs.
- Ensure that the questions are drawn more or less equally from the whole course e.g. with 80 questions and twelve teaching weeks there should be 6-7 questions on each week's work.
- Each question set should include some very easy straightforward questions, some tough ones and a range of in-between ones; collectively the set will then have good discriminating power.
- When planning a second exam using a pre-existing question bank, check that new questions are written to take new material into account.
- Read old questions before setting a new exam. Some questions are not so clear to their composer a year later!
- Ensure that the questions relate to the learning oucomes for the module. Where appropriate, check that the questions assess a range of skills (such as manipulating units, interpreting graphs etc.) as well as testing the facts and ideas that form the knowledge base of the module.

### *Reference*

Angseesing, J. 1989. Open-ended computer-marked tests. Teaching Earth Sciences 14, 17-19.

# CASE STUDY TWO: World Wide Web based formative assessment in psychology

Tom Buchanan

# World Wide Web based formative assessment in psychology


### *Author and contact:*

Tom Buchanan
Division of Psychology
University of Sunderland
The Business School
St Peters Campus
St Peters Way
Sunderland
SR6 ODD
UK

### *Email:*

tom.buchanan@sunderland.ac.uk


### *Web URL:*

http://www.sunderland.ac.uk/~hs0tbc/psychome.html


### *Key-words:*

Automated, formative, assessment, World-Wide Web



### *Outline:*

When faced with large numbers of students and little time in which to teach them, it is often difficult to provide timely feedback which students may use to improve their performance. This case study describes an automated WWW-based formative assessment package which is both diagnostic and prescriptive in nature. Integrated with the course text, it is used in a test-study-retest cycle which students may repeat as appropriate. Students do sets of multiple choice questions on-line. These are then automatically marked and a feedback page is generated. The feedback advises students on appropriate reading they can do to fill any gaps which have been found in their knowledge.


### *Context:*

Psychology at Sunderland is taught within a semesterised modular framework. Within each 15-week semester there are 12 teaching and 3 assessment weeks. The activity described here was initially used in a Level 1 Introductory Psychology module [PSY111] with approximately 300 students. It has since been used in other modules.

### *Description*

The computerised package, PsyCAL (Psychology Computer Assisted Learning) described here is based around short sets (10 to 20) of multiple choice questions. Integrated with recommended texts for the course, it is used as part of a test-study-retest cycle. The main aims of the exercise are:

- To allow students to assess the extent to which they have mastered key areas of curriculum

- If mastery is not complete, to indicate specific remedial work which can be done
- To highlight areas of reading students should concentrate on
- To allow students to work at their own pace and in any location they prefer
- To increase efficiency of delivery through automatic provision of feedback
- To contribute to a culture of student autonomy and student-centred learning

### *The System*

The package is Web based[1] and can be accessed at any time from any location on or off campus -anywhere there is an Internet-connected computer. On accessing the PsyCAL home page, students select a set of multiple choice questions which is presented on screen as an HTML form. Students answer these questions on-line by clicking the appropriate radio buttons, and then select a "Submit Answers" button. The answers are then immediately marked and a feedback page appears on screen.

The feedback page tells students how many questions they answered correctly. It also lists the questions they got wrong. The correct answers are not given, as it is felt that this would encourage rote memorisation and surface learning. However, for each wrongly answered question a reference is supplied to a section of the recommended course texts where the answer may be found. Students therefore make a note of the questions they answered wrongly, go away and seek out the answers in the recommended material, then attempt the exercise again at a later date. They are advised to repeat this process until they can successfully answer all the questions. It is believed that this approach results in a more active engagement in the learning process: students must seek out, rather than just passively receive, the information they require.

### *Implementation*

At the start of the course, students were provided with a manual outlining how to access and use the package. Open tutorial sessions on how to get started were made available for anyone who felt the need: few (perhaps 10%) came along to these. Those who did required almost no help. During the course, students reported few problems with using the package.

Within the 12 week teaching period, three specific weeks (4, 7 and 11) were designated in which students were required to use the package unsupervised, at their own pace. In these weeks, they were asked to complete sets of questions relating to material just covered in the lecture course. Students were encouraged not to restrict their usage to these weeks: sets of additional revision questions were also made available which students could use as they wished.

### *Resource Implications*

This is a relatively inexpensive activity. The major investment was the staff time taken to develop the software (several weeks) and the various question sets (comparable to designing any multiple choice test). Once this is done, however, the exercises can be used year upon year with only occasional updating when required.

Rather than attending special classes at set times, students were told to do the exercises in whatever time and place was most convenient to them, using the open-access computing facilities around the campus.

[1] It is in fact a collection of interacting HTML pages and CGI scripts written in the language Perl, all hosted on the University's Web server.

Because the system is fully automated, no staff input or supervision is required when students are using the package.

As the package is text-based and accessible via the WWW, expensive hardware and specialist software are not required: exercises can be accessed via any Internet-connected computer running any standard browser software.

### *Student Assessment*

Repeated automated assessment is the key characteristic of this activity, as described above. The purpose of the assessment is to provide feedback to students which they may use formatively. Given that this is a tool for learning, no attempt is made within the procedure to summatively assess students.

A review is currently being undertaken of the extent to which use of the exercises contributes to achievement of the learning outcomes for the relevant modules. However, students have reported that it was at least as valuable any other teaching and learning method in this respect.

### *Evaluation*

The system seemed popular with students. In the first year (1996-97) that it was used in the module PSY111, the system was accessed a total of 1809 times. In the second year (1997-98) this rose to 2310. In an end-of-module feedback questionnaire, students were asked to report the extent to which they had used the system. 93% of respondents indicated that they had done all of the set exercises, and 74% said they had done at least one of the extra revision exercises.

In the same questionnaire, students reported their experience of various aspects of the course and of the PsyCAL exercises in particular. They rated the perceived usefulness of the main teaching and learning methods employed (lectures, seminars, textbooks and PsyCAL). Results indicated a reasonably high level of satisfaction with all techniques, and a repeated measures ANOVA indicated that there were no significant differences between the techniques in mean satisfaction.

97% of respondents replied affirmatively to a question asking if they would like to use the package in other modules. Comments from students were very positive, indicating that users were satisfied with the system and would like more questions to be available.

The major problem which users have encountered is in gaining access to computers on which the package could be used. While the intention was that people would simply do the exercises at a time and place convenient to them, the high pressure on open-access computing facilities around campus meant that some students experienced difficulties. Several people requested that time be reserved in computer classrooms for them to use the package, and in future this will be done.

### *Key advice*

- Use a commercial system rather than developing your own software - this is where the major time commitment was incurred and most problems were encountered. Software (such as Question Mark Web) to support on-line assessment is increasingly becoming available, and the functionality of the system described here can easily be replicated using such a package.
- If there is a lot of pressure on computing facilities, time should be reserved in computer classrooms for students to do the exercises (although the option to work at their own pace, rather than in a formal class, should remain).

# CASE STUDY THREE: Formative assessment in a basic geographical statistics module

Dan Charman and Andrew Elmes

# Formative assessment in a basic geographical statistics module


### *Author and contact:*

Dan Charman & Andrew Elmes

Department of Geographical Sciences

University of Plymouth

Drake Circus

Plymouth

PL4 8AA

UK

### *Email:*

dcharman@plymouth.ac.uk, aelmes@plymouth.ac.uk



### *Key-words:*

statistics, objective testing, formative assessment, student performance, student evaluation



### *Outline:*

This case study describes the content of, and rationale for, the introduction of CBA for formative assessment on a first year module in geographical data analysis and then goes on to evaluate student opinion and exam performance in response to this. Questionnaires showed that students were generally positive about the assessments, and overall module evaluations improved. Student interest in the subject matter also increased. Comparison of summative examination performance before and after the introduction of CBA showed an increase in average marks, with particular improvement shown in the range of marks 35-50%. These results show that CBA can be effective at delivering improved student performance, as well as being popular with students.


### *Context:*

'Techniques of Geographical analysis II: data analysis' is a compulsory semester length module for all first year students in geography and earth sciences at the University of Plymouth. Currently up to 275 students take this module which is taught using a series of practical exercises supervised by postgraduate demonstrators. The practical exercises are based on a handbook and are supplemented by lectures to introduce topics and solve problems for previous exercises. In the past, assessment was based on a portfolio of ten practical assignments and two in-module OMR based tests. However, assessment of portfolios led to inadequate feedback to students and late return of work, due to the sheer volume of material to be marked by staff. To address this problem, the portfolio assessment was replaced by a series of weekly CBAs.

### *Description*

The CBAs use Question Mark software to set short (10-15 minute) tests based on questions selected randomly from stratified question banks covering 1) the answers to the practical exercise, 2) interpretation of further statistical data and 3) statistical concepts. Students receive feedback specific to each response wherever possible. There is no time limit for each CBA to allow students to gain maximum benefit from feedback, and tests are subsequently available as a revision resource. Student answers are logged and can be used to compile marks and check each individuals progress. Students also sit an end-of-module examination similar to the previous in-module tests.

### Resource Implications

The main development cost was the time to write the questions and feedback and to implement these on the software and network. This was only achieved with the assistance of a small institutional grant which employed a research assistant for 2 months. We also had to purchase the software which for Question Mark Designer and Network Guardian was just over £1,000, although this cost has subsequently been defrayed by further users. Running costs are minimal except when something goes wrong when considerable time may be needed to sort out the problems. This time is still much less than the time previously spent on manual marking. Technical backup with the network has also been essential to the smooth running of the CBA and this is helped considerably by a dedicated departmental server and computing room so that control of the system is assured.

### Student Assessment

The two in-course tests which had run previously were assembled from a data bank of questions. When these particular tests were replaced with an end-of-module examination, the format was retained and the examination was based on the same bank of questions and ran for the same length of time as the two tests combined. The content of the examination was thus kept the same as in previous years in order to provide an assessment of the effect of the introduction of the CBAs on student performance. A comparison of the results from before and after the introduction of the CBAs (Figure 1) shows an improvement in average marks from 54.1 (s.d. 12.3) to 56.8 (s.d. 12.9). However, this small mean increase masks the detail of change which was not uniform across the spectrum of student performance. Minimum and maximum marks stayed approximately the same at c.25% and 85-90% respectively and there was little change in the number of students obtaining marks of less than 35% or more than 75%. The biggest changes were in the numbers of students who were on the border-line of failure (pass mark was 40%) or who achieved marks between 40 and 50%. This is strong evidence that student performance can be enhanced by the introduction of CBA as a learning tool for formative assessment. It appears to be particularly beneficial to students who are struggling with the subject and who otherwise might fail or barely pass the examination.There is still a small group of students who fail the module and another group of very high achievers who do not appear to benefit from the change in assessment.

### Evaluation

Student opinion was obtained by two questionnaires:

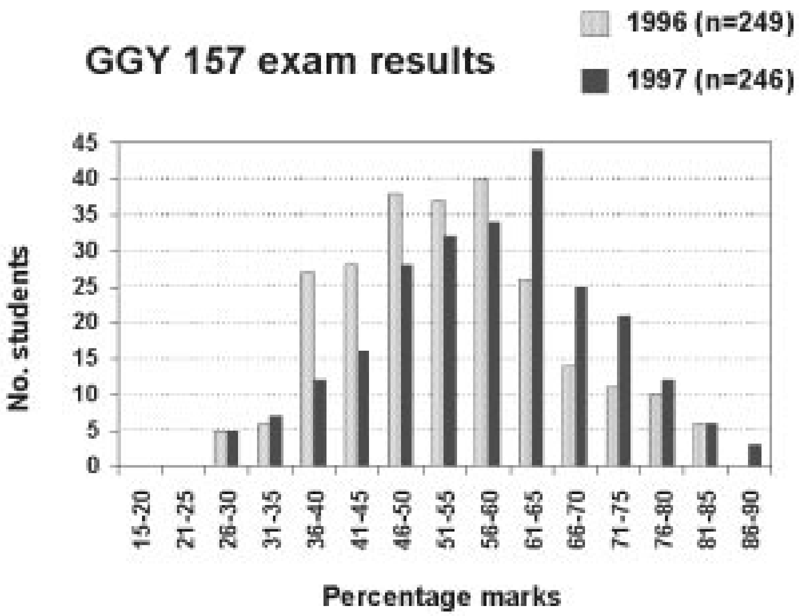
- (i) A questionaire on CBA in this module

Student response was generally positive, with 64% agreeing that the CBAs were a good way of learning, 68% agreeing that the feedback was adequate and relevant and 56% agreeing that the CBAs were an improvement over other forms of assessment. There are some problems of making marking sensitive to the idiosyncrasies of student response on text questions but 85% of students considered the assessments fair. This positive response is similar to that found in other studies on the introduction of CBA for summative assessment (e.g. Pritchett & Zakrzewski, 1996).

- (ii)The standard module evaluation questionaire which is issued for all course modules

A comparison of student evaluation of the module with previous years' showed that overall satisfaction scores improved markedly with the introduction of CBA. When asked to agree or disagree ( on a scale of 1=strongly agree to 5=strongly disagree) with the statement 'Overall, I am satisfied with the quality of this module', average scores improved from an average of 3.01 in the years 1994 to 1996 to 2.34 in 1997. This suggests that the change in assessment strategy had a major effect on student satisfaction since other aspects of the teaching style were basically unchanged. Perhaps more noteworthy is the improvement in student interest in the module. When asked to agree or disagree with the statement 'I found this module stimulating and interesting' scores changed from 3.64 (1994), 3.77 (1996) to 2.76 (1997). This is an unexpected but very welcome benefit of the change in assessment procedure as it is generally difficult to stimulate student interest in modules which deal with data analysis techniques.

**Figure 1. Comparison of exam performance before (1996) and after (1997) the introduction of CBA for formative assessment.**




- Staff evaluation

From the staff point of view, the module management has become a little more complicated and, to some extent, more distant from the students but it has improved the quality of assessment and saved time in only the second year of use. All the bugs are still not solved and this does cause some frustration from time to time. Also future questions will need to be added in further years and this will take a little time although it is a straight-forward task once the software is mastered. Overall, the introduction of CBA in this module was an effective way to combat the problems with formative assessment of large numbers of students.

### *Key advice*

- Try CBA for formative assessment initially. It imposes less stress on both students and staff
- It helps to have some start-up assistance in the first year to assist in developing and implementing the materials
- Make sure you have adequate advice on and control of the network where your assessments will be delivered
- Don't assume you will save time right away - it may not happen in the short term! The savings acrue later

### *References*

Pritchett, N. & Zakrzewski, S. (1996). Interactive computer assessment of large groups: student responses, Innovations in Education and Training International, 33, 242-247.

# CASE STUDY FOUR

## Mastertutor - telematic delivery of student-centred tutorials in problem solving

Phil Culverhouse and Cheryl Burton

# Mastertutor - telematic delivery of student-centred tutorials in problem solving


### Author and contact:

Phil Culverhouse[1] & Cheryl Burton[2]

[1]School of Electronics, Communications and Electrical Engineering.

[2]Academic & Information Services

University of Plymouth

Drake Circus

PL4 8AA

UK

### Email:

phil@cis.plym.ac.uk, cburton@plym.ac.uk



### Key-words:

telematics, student-centered tutorials, problem-solving support, world wide web



### Outline:

Mastertutor is a World Wide Web based tutorial shell that assists in the formation of problem-solving strategies and design competencies. The system sets a problem, provides an information resource, requires a written solution by the student, ascertains the student's solution via a questionnaire, assesses the solution and defines what was expected of the student through a solution discussion. On completion of the tutorial, students may debate the problem and solutions on a bulletin board system controlled by Mastertutor; answers may be provided by both peers and tutors. Students using Mastertutor benefit from wider reading, automatic assessment and staff contact through solution discussions.


### Context:

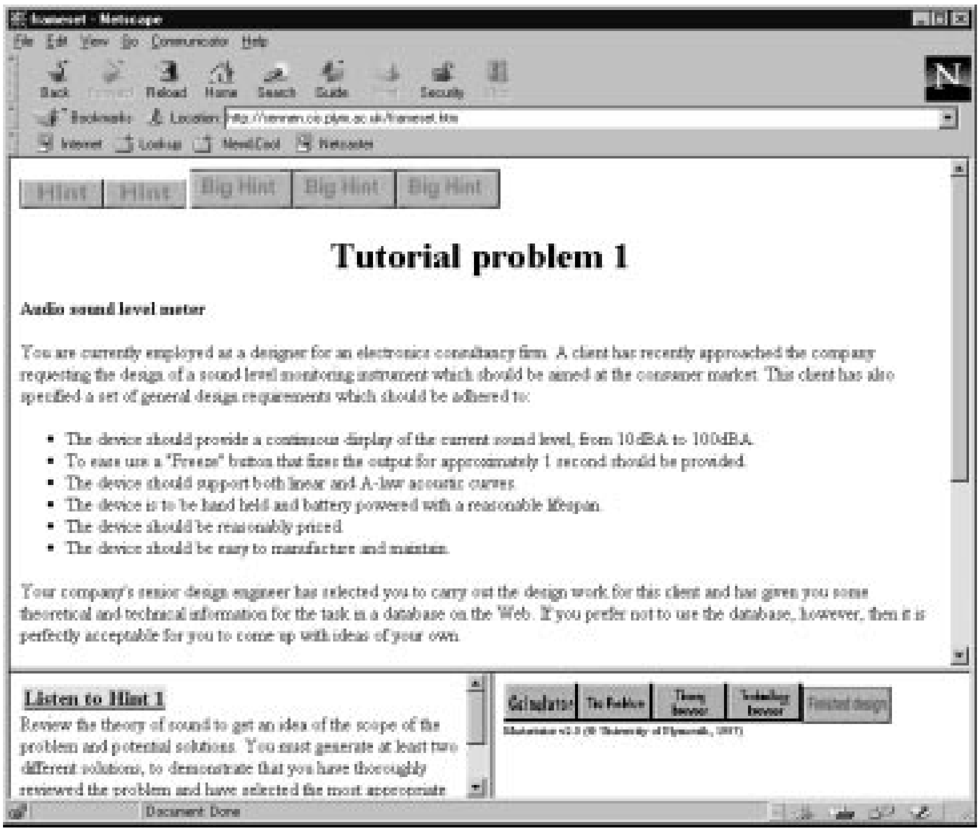
The system has been evaluated in the laboratory at Plymouth with over 50 professional and undergraduate students. It is now undergoing extended live trials with students on the final year B.Eng. and B.Sc. Electronics Engineering degrees. Tutorials are also being developed to assess the technique in geography, psychology and geology.

### Description

Mastertutor draws its inspiration from the music master-class, where (i) a student plays a piece of music for the tutor, (ii) the tutor assesses the performance and then (iii) plays the piece back to the student. The student compares her or his performance with that of the master and improves by identifying the differences. Following this master-class analogy, (i) the Mastertutor system sets a problem, provides an information resource, requires a written solution by the student, (ii) ascertains the student's solution via a questionnaire, assesses their performance and awards them a mark and (iii) finally defines what was expected of the student through the presentation of valid solutions. In addition, students still having difficulty or who would like to contribute to a general discussion on the topic with their tutor and peers may post questions or comments to a discussion group in the form of an on-line bulletin board system controlled by Mastertutor.

The information resource can be looked at in any order and for however long the student may wish; in other words, there is no prescribed route through it. This approach mirrors the real-world method of investigation and promotes traditional academic learning. This resource was carefully chosen to contain more information than is needed to ensure a serial search will not reveal the solution. The student can get help from the database in the form of 'Hints'. Each Hint appears in both spoken and textual forms. The voice of the tutor is recorded for the spoken form to maintain the link between student and tutor and to reduce feelings of student isolation. Hints can be of two types: Hints say what the next step is while Big Hints give part of the problem solution. More marks are deducted for the latter than the former, although once a particular Hint or Big Hint is chosen the student can listen to it repeatedly without further penalty. Figure 1 shows a Mastertutor problem page.

**Figure 1. A Mastertutor problem page**




The assessment is in three parts: a solution mark (from the solution questionnaire), a Hint mark (deducted) and a strategy mark (how the student approached the problem solution). For student-performance analysis, the tutor receives for each student a profile giving details of their marks, search strategy, Hint usage and questionnaire answers.

The learning feedback cycle is closed through the solution discussion whilst the problem is still fresh in the student's mind.

### *Resource Implications*

Database preparation time is difficult to generalise, but seems to be approximately 2 minutes per A4 page of text and/or graphics. Assuming that a tutorial problem has to be constructed from scratch, it is estimated that its solutions, accompanying Hints and assessment questionnaire consume 20 hours of effort.

When running Mastertutor live on the world wide web, the average information demand from the web-server is 50KB per 3 minutes per student. If a file server can support only 4 access database transactions per second then it can support 720 student accesses per minute.

Mastertutor reduces staff contact time by filtering out those students who require additional remedial support. In a laboratory study only 1 student in 24 required direct tutor contact to explain the tutorial and its solutions, resulting in a 15 minute discussion. Normally 6 hours of staff contact and 6 hours of marking would be required for a group of this size (assuming 4 students per one hour tutorial and 15 minutes per student solution marking time). Mastertutor therefore saves 11 hours and 45 minutes.

### *Student Assessment*

The authors have studied 21 professional and student engineers solving a concept design problem using a development version of Mastertutor (Culverhouse et al. 1992). These studies highlighted differences in expert and novice problem-solving strategies. Further studies with Mastertutor using 24 student subjects confirmed the simplicity of the database browser navigation method, the great value of the solution discussions and the general acceptance of the automatically-awarded mark. In addition, only 1 student of the 24 found the interaction for a two hour period with the computer TV display screen really arduous.

All 53 evaluation questionnaires analysed to date signal significant positive responses to the principal features of Mastertutor and to the value of the tutorials. Table 1 summarises their responses.

**Table 1. Summary of subject questionnaire responses**

| Evaluation equation | Summary response from subjects |
|---|---|
| How satisfied generally with Mastertutor were you? | Very satisfied |
| How satisfied were you with your final score? | Indifferent |
| How satisfied were you with the theoretical information? | Mildly satisfied |
| How satisfied were you with the practical information? | Indifferent |
| Did you have to supply much additional knowledge? | A minimal amount |
| How easy was the problem to solve? | Satisfactory / fairly difficult |
| To what extent was Mastertutor helpful in your design? | Considerably helpful |
| Did you find the discussion helpful? | Considerably helpful |
| Would you like to solve more problems using Mastertutor? | Yes |

Although conclusive proof is not yet available (long-term studies are continuing), this feedback indicates that the student learning experience is positive. Students may well be improving their problem solving skills just by doing more problem solving coupled with the timely feedback of solutions. They may be increasing their domain knowledge since they were active in their reading, rather than passive.

The solution discussion was felt by many to be of great value, as indicated by the comment below from a student who recently took a Mastertutor tutorial: "*an excellent and totally different approach to solving a problem. I would like to try different problems to improve my design method*".

### *Evaluation*

Mastertutor offers a number of benefits to both staff and students. It has also some drawbacks.

- Staff benefits

Mastertutor allows the provision of more tutorials without additional burdens on staff. It reduces the numbers of students seeking assistance with the tutorial when dealing with large classes. It encourages student-centered learning and has a particular strength in formative assessment. Mastertutor is easy to use and tutorials may be set which accommodate the student ability range. The support tools provide rapid development of new tutorials and the on-line analysis package provides a detailed report on each student session. It can deal with large classes. Each student will receive the same learning experience.

- Student benefits

Learning is self-paced and students can choose to do a tutorial when it suits them. Information is on-line, so students don't have to spend hours looking for books. The tutor is effectively on-hand through the Hints and solution discussion. Feedback is instantly available on the student's own performance. Mastertutor provides well-thought through solutions assuring a high quality learning experience. There is no peer pressure. Every student is treated the same, no favourites, personality clashes, no prejudices (colour, sex), which can all be issues in face-to-face tutorials. Personalised help is available through the discussion group or the tutor.

- Drawbacks to staff

When holding traditional small group tutorials, (an alternative to The Master-tutor approach) staff can reduce their preparation burden by giving highly interactive tutorials which rely on the tutor's skill in pressing students for concepts and solutions. In these situations staff do not have to have detailed solutions to hand, as they can rely on members of the tutorial group to voice these solutions. However, when Mastertutor replaces the tutorial, the preparation must be more detailed. In particular the solution discussion may be time consuming to complete in comparison to a normal tutorial, due in part to the need to make detailed public statements of solutions which could cause debate with one's colleagues. This debate can be defused by involving peers in the preparation of the solution discussion.

- Drawbacks to students

There is less personal contact with tutors.

### *Key advice*

- Consider using Mastertutor in addition to existing personal or group tutorials rather than as a replacement.
- Select tutorial problems that have more than a single satisfactory solution.
- Ensure that the discussion commences with 'how I tackled the problem'.
- Each solution should be discussed in terms of its benefits, its limitations and particular difficulties in implementation.

### *References*

Culverhouse P.F. , Ball L. & Burton C.J. (1992) "Desmate: A Tool For Tracking Engineering Design In Action", in Design Studies Jan 1992, vol 13 no.1 pp-54-71.

Culverhouse P.F. & Burton C.J. (1998) Mastertutor: a tutorial shell for the support of problem solving skill acquisition, in BITE - Bringing Information Technology to Education: Integrating Information & Communication Technology in Higher Education, Eurelings A. (ed.) Kluwer Amsterdam ISBN 90 268 320 60.

# CASE STUDY FIVE
## Mathletics - tests for diagnostics and assessment in mathematics for science students

Martin Greenhow

# Mathletics - tests for diagnostics and assessment in mathematics for science students

***Author and contact:***

Martin Greenhow

Department of Mathematics & Statistics

Brunel University

Uxbridge

Middlesex UB8 3PH

UK

***Email:***

mastmmg@brunel.ac.uk

***Web URL:***

http://www.brunel.ac.uk/~mastmmg/

***Key-words:***

mathematics, diagnostic, formative, summative, Question Mark

***Outline:***

The Mathletics program comprises a suite of mathematics tests for diagnostic, formative and summative assessment in commonly-taught areas of number, algebra and calculus from GCSE to year 2 undergraduate level. The aim is to provide an assessment environment where topic question libraries can be combined in various ways (pentathlon, decathlon etc) and immediate feedback given. Each test starts with information and revision screens. Much of Mathletics was written by final-year project students using Question Designer.

***Description***

Mathletics is designed to:

- Diagnose freshers' mathematical abilities and preparedness; inform the student; inform the lecturer/tutor; get the student to revise their A-level notes before the first-year modules start "hotting-up"; suggest corrective action.
- Offer a substitute for some worksheets for many modules. Each test may be repeated, offering a different choice of randomly-selected questions, so that practice for speed and accuracy is available, with instant feedback. Lecturers can monitor students closely and take corrective action, either individually, or as a class revisiting problem topics.
- Provide revision tests for students e.g. the differentiation decathlon.
- Provide formal (invigilated) exams and class tests. Here feedback is suppressed, and questions may be answered in any order with questions being revisited, until the student is satisfied and presses "finish" or runs out of time. The answer file is then securely written, and the student is given his/her mark and informed of, and given feedback on, all incorrect answers.

- Provide a set of tests for admissions tutors to assess the current mathematical ability of a wide range of students including those with unconventional previous education, transfers from other courses, etc.
- Provide an informal resource in basic arithmetic and algebra for students on a wide range of courses where numeracy is vital but may not be formally taught.

### *Student Assessment*

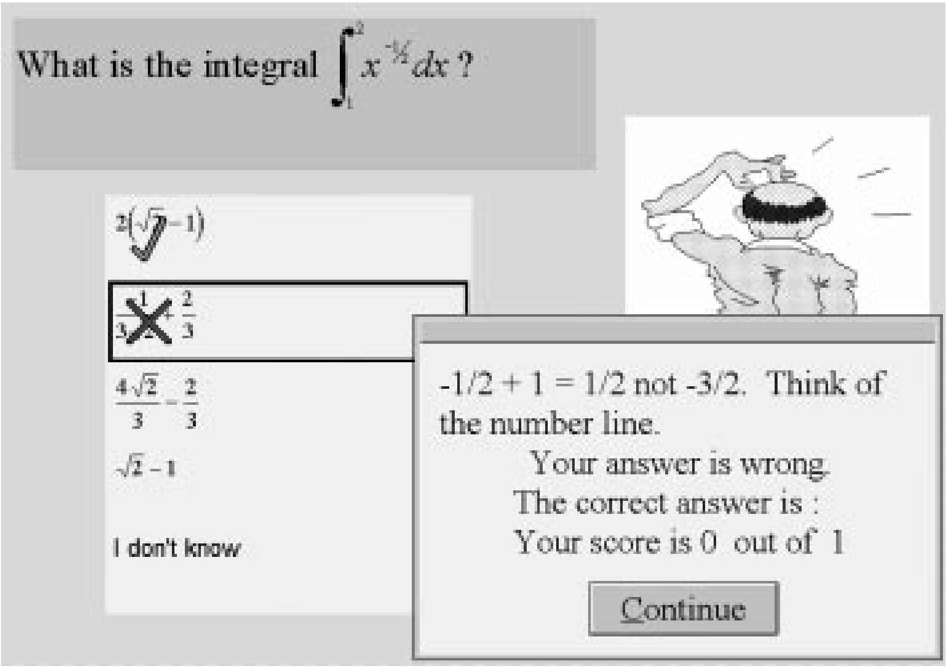
The questions are mostly multi-choice with distracters (displayed in a random order at each run) chosen by common mal-rule application, with appropriate feedback (Figure 1). Thus a student will not only know that he/she is wrong but will be told why. Multi-choice answers are also easily analysed by the lecturer. Other types of questions are also used such as hot-spot (point to the mistake in this mathematics - see Figure 2), numeric (what is the value of this integral?), multi-choice (which of the following apply to this function?), and simple yes/no answers. The questions can allow multi-media buttons, which might, for example, display a calculator only when the question setter thinks it should be available, or more advanced facilities.

At Brunel University, Mathletics is used for diagnostics of 500-800 science or technology students, and as assigned tests for 90 mathematics and 60 foundations of science students plus much other informal use.

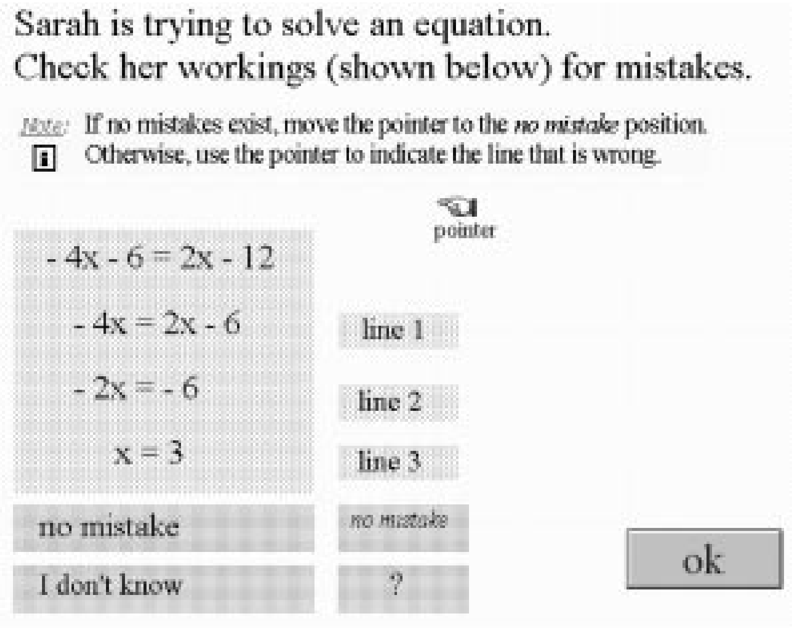
### *Evaluation*

Feedback, collected via on-line questionnaires and informally, has been extremely positive; students require no training to use the software, data collection is robust and feedback to students is viewed as accurate. Display problems and a few mistakes in version 1 have been corrected. Mathletics is under continuous development; currently we are developing tests for elementary statistics, OR and Laplace transforms, as well as diagnostic tests which are responsive to student knowledge rather than based on blanket topic coverage.

**Figure 1. Typical question with brief feedback from Mathletics**



### Figure 2. Typical hotspot question from Mathletics




#### *Key advice*

- Do award some module marks for Mathletics test; otherwise only well-motivated students will actually complete the tests, whereas under-motivated students will think "What is in it for me?" and will fail to learn from the tests.
- Do ask your students to write down problem questions and discuss them in subsequent tutorials.
- Do try the software first - logged in as a student. This will check that the read/write permissions have been correctly set and that you can actually read their answer files.
- Do check the students are not cheating; group work can (perhaps should) be sanctioned, but using aids such as Derive to do the problems, and simply inputting the answers is pointless.
- Do not eliminate paper-based tests and worksheets completely; if you do, students may have had little experience in solving problems and laying out mathematical arguments when faced with a blank sheet of paper.
- Do not expect Mathletics (or CAL systems in general) to rank order your students correctly. Good students may get bored and stop at 70%, whereas weaker students often retake tests until they get 100% simply to get marks as a buffer against anticipated poor exam performance (but without learning much more).

# CASE STUDY SIX
# A multimedia tutorial shell with qualitative assessment in biology

M.J. Hall, D.J. Robinson,
G. Tucknott and T. Carlton

# A multimedia tutorial shell with qualitative assessment in biology


### Author and contact:

M.J. Hall, D.J. Robinson, G. Tucknott and T. Carlton
Biology Department
Open University
Milton Keynes
MK7 6AA
UK

### Email:

m.j.hall@open.ac.uk, d.j.robinson@open.ac.uk, g.m.tucknott@open.ac.uk, t.f.carlton@open.ac.uk



### Key-words:

Generic shell, multimedia, qualitative assessment, independent learner, navigation


### Outline:


The project is developing methods to produce multimedia tutorials relatively quickly and cheaply, using a generic software shell suitable for any subject area. Tutorials presented in the shell will provide the student with a structured learning experience that will allow their initial knowledge level or their knowledge acquisition and progress to be qualitatively and quantitatively assessed. Where areas of weakness are revealed by the assessment, students will be advised to study particular parts of the tutorial in order to improve their understanding.


### Context

We are developing the shell in the context of a project to produce The Human Brain CD-ROM, a multimedia tutorial on the human nervous system, for the Open University course SD206 Biology: Brain and Behaviour. This level 2, 60-point course attracts about 800-1000 students each year, about half taking a science and half a psychology degree. A small-scale prototype containing a single introductory section (the 'Overview'), was sent to students in 1997. A more advanced version, though still without the Concept Test assessment component (see below), containing seven sections, is being sent to 1998 students. The final version of The Human Brain, which is expected to have 10-12 sections, will be completed in 1999. At present, The Human Brain is optional material for students. From 1998, all students are obliged to use a PC, so it may be decided to make the CD-ROM compulsory in 1999. The shell developed for The Human Brain is now being used to develop a number of other biology tutorials, but the aim from the beginning has been to produce a generic shell that can be used for any subject matter.

### Description

Our tutorial shell is based on the 'Scholar's Desktop', produced by the Biodiversity Consortium (Davies, 1995). We have modified it to include a qualitative assessment component (see below) and to provide a more defined tutorial structure which acts as an aid to navigation. The software shell is generic in the sense that it simply provides the functionality required to present multimedia materials and link them together. The contents, which can be on any subject, are held as separate files external to the shell. They are put into the shell, and the specific links required between them are set up, using a simple, bespoke, text-based programming language called DeskTop Code. We are in fact developing a number of biological tutorials based on this shell, though so far *The Human Brain* is the most complete and extensive. These tutorials are designed for independent learners either in a distance-teaching situation or taking a course in a conventional institute of Higher Hducation.

Both the shell and any tutorials produced in it are very flexible. The style and 'feel' of a tutorial can be varied by, for example, not using all the levels or functions available in the shell, or changing the background, or the colour and shape of the buttons. Once a tutorial is produced there are several ways for the developer to update and extend the materials it contains. For example, individual graphics, sound clips or video clips can easily be replaced with an updated version and a 'News' option provides a text file containing extra information, commentaries etc., which can be added to or updated at any time. Individual assessment questions can also be replaced relatively easily. It would be possible to make some of these updating options available to the teacher using the tutorial, if this were thought desirable.

### *Resource Implications*

By the time it is completed, a huge amount of resources (both money and academic time) will have been put into modifying the Biodiversity shell and producing The Human Brain tutorial. But once the shell is completed, the cost of producing new tutorials will be far less. They could be produced relatively cheaply in terms of the time required to write the academic content and code the materials into the shell. The main cost is likely to be in acquiring high-quality graphics: producing them from scratch is expensive and re-using already published images requires the payment of copyright fees.

### *Student Assessment*

The shell offers two kinds of assessment: Quick Test and Concept Test. Quick Test consists of multiple choice questions, with feedback for incorrect answers. Students are given an accumulated score as they progress through a section and answer more questions. A student is able to call up Quick Test at any time, close it down and open it again, without loss of their accumulated score. If the student leaves a section and later returns to it, the Quick Test scores are retained. Students only get questions on subjects they have already studied in the tutorial but each question is called up randomly from the bank of possible questions.

We are developing the Concept Test component in collaboration with colleagues at the National Centre for Software Technology in Bombay. The two forms of assessment work similarly in terms of the way students access and answer questions and the way questions are linked to particular screens. Concept Test and Quick Test do, however, have major differences. Each section of the tutorial has a number of associated learning objectives, which includes one or more 'concepts'. For example, if one of our objectives is 'You must decide what to eat for breakfast', then the concepts might be 'decide', 'eat', and 'breakfast'. In Concept Test, each question is written to test a particular concept, which in turn tests the student's understanding of the related learning objective. When they have finished working through the questions, students are provided with a separate score for each concept, so that they can see where their understanding is poor, and be invited to study a remedial tutorial for any concept they feel they are weak on. The remedial tutorial takes the student through the relevant materials again but is tailored precisely so that only screens necessary to understand that particular concept are included. A 'save' option allows users to save and reload score information for both Quick Test and Concept Test between sessions.

As well as standard multiple choice questions, we are planning to incorporate a number of variations into Concept Test, including questions where students are:

- asked to identify an image
- presented with an image and asked to click on a particular area

- presented with an image with blank labels, plus a list of labels which can be selected and placed on the image as appropriate.

In future, further question types may be added, e.g. drag and drop exercises in which items can be sorted into categories; questions that require answers to be entered as text etc.

In our tutorials, both Quick Test and Concept Test are used by the student entirely for self-assessment. However, since the computer keeps the score and allows the student to save their results, it would be a simple matter for a teacher to collect student scores and use them for formal assessment.

We also intend to add a further option to the software so that Concept Test can also be used as a pre-test to assess the level of student knowledge and understanding before they study the tutorial. In this option, the linkage between questions set and screens studied will be 'turned-off' so that all questions are available immediately. Such a pre-test will allow the tutorial to be used e.g. to bring a group of students coming from a variety of backgrounds to be brought up to the same level before they start a course.

### *Evaluation*

The Overview was sent out to students taking the course in 1997. Students were asked to complete a questionnaire, whether they used the CD or not. Of the 83 students who returned the questionnaire, three fifths had used the CD. All those who had not used it said that this was simply because they did not have access to a suitable PC. The feedback received was almost entirely favourable (Kirkwood, 1997):

- Most students (92%) found the three-layered structure fairly or very helpful as a means of guidance through the materials.
- Most students found the CD easy to operate (86%) and enjoyable to use (84%)
- Most students thought it was better learning about the nervous system on CD than in a book (48% said it was better, 34% said it was better for some topics, 10% said it was about the same)
- Students used the CD as an initial tutorial (66%), for revision (36%), and as a reference source (38%)
- Most comments were enthusiastically favourable and many students said they would welcome more CD-ROM materials.

Later this year, a similar questionnaire will be sent to SD206 students with the fuller version of the CD-ROM, when it is hoped the response rate will be greater.

### *Key Advice*

- Using a shell reduces the academic and programming time needed to develop a multimedia tutorial considerably, but not the resources needed to produce high-quality graphics.
- Producing multimedia tutorials, even with a shell, takes more time and resources than producing single-medium teaching materials.
- A means of navigation that is simple, obvious and consistent, and which does not let the student get lost while giving them the opportunity to explore, is important if students are to achieve the learning objectives.
- The system needs to be flexible so that tutorial styles can vary and so that the teacher can adapt it to the needs of their own courses.
- Usability of the system must be assessed by the user not the developer.

### References

Davies, P. (1995), Biodiversity Consortium, in: Teaching and Learning Science Case Studies, pp. 7-12, TLTP.

Kirkwood, A. (1997), SD206 The Human Brain CD-ROM survey 1997. PLUM Paper No. 98.

Open University Institute of Educational Technology, Milton Keynes.

# CASE STUDY SEVEN
# The BOSS System for On-line Submission and Assessment of computing assignments

Mike Joy and Michael Luck

# The BOSS System for On-line Submission and Assessment of computing assignments


### Author and contact:

Mike Joy & Michael Luck
Department of Computer Science
University of Warwick
Coventry, CV4 7AL
UK

### Email:

M.S.Joy@dcs.warwick.ac.uk

### Web URL:

http://www.dcs.warwick.ac.uk/Boss/



### Key-words:

online submission, automated testing, programming



### Outline:

Practical computing courses which involve significant amounts of programming continue to suffer from increasing student numbers. This makes their delivery and management more difficult to achieve effectively with the available resources. One solution to this problem is to develop methods for automating the submission and testing of student programs to support the marking effort and to enable the division of marking tasks among several individuals while ensuring consistency and rigour throughout. We have developed such methods in our system, called BOSS, and have successfully deployed different versions of it on several courses over a number of years. Here, we describe the original system and its recent enhancements, and discuss the benefits it has provided us with, both in terms of administration and in improving the learning process.


### Context

The effects of ever-increasing student numbers continue to impact on the quality of course delivery in both further and higher education. This is especially true for computing courses in which there is a high practical programming component, and which are assessed by programming assignments. With large numbers of students, many lecturers are faced with the stark choice of investing vast amounts of time and effort in marking, through the separate processes of retrieving student programs, compiling them, running them and testing them manually against sets of test data, or of simply visually scanning a program and assigning a mark to it on the basis of a cursory assessment. The first of these options is becoming impossible due to the increasing demands on lecturers, yet the second is hardly acceptable.

By developing techniques for automating the submission, compilation and testing of student programs, we can support the process of marking, and enable marking tasks to be divided among several individuals while maintaining rigour and consistency. Not only can this stem the tide of an increasing workload, it can improve the learning experience available to students (in providing them with facilities for immediate and effective feedback, for example,) and also enable other administrative tasks to be automated as part of a coherent approach to full course management.

We can distinguish between courses which introduce specific programming languages, and those in which broader aspects of software engineering, such as techniques for design, methodologies, and analysis of algorithms are taught. It is not typically the purpose of introductory programming courses to cover such issues, which are often regarded as separate areas (Finkelstein, 1993). In considering the requirements for an automatic submission system, we focus specifically on just such introductory programming courses.

For these reasons, a programming assignment should be carefully and accurately specified. Without such a precise specification of the task, the assessment of programs can become significantly more difficult, as conformance to the specification may not be easy to judge. A tight specification, however, allows the submitted program to be tested against suitable test data, so that the output of the program can be compared with the expected output for each set of data. The requirement that solutions must conform to specification thus serves the dual purpose of enforcing good practice and enabling a measurable assessment of the program.

For several years, we have been using such a system that allows students to submit programming assignments on-line and to run those programs against test data (Luck and Joy, 1995). The system we have running is in its fifth year of use, albeit in different versions. It has been deployed on several courses, including those covering Pascal programming, UNIX Shell programming and C++ programming, each course attracting up to 200 students.

### *Description*

The BOSS system for automatic submission of assignments (Joy and Luck, 1995) comprises a collection of programs, each of which performs a different task contributing to the overarching goal of effectively managing the process of submitting programming assignments on-line. BOSS is designed specifically for courses with large numbers of students, assessed by means of programming exercises.

The programs offer the following functionality:

- Students may submit programs on-line by means of a user-friendly program that conducts a dialogue with the student to ensure that the correct submission is made. The program is stored and simple checks are carried out so that the lecturer can subsequently test and mark it.
- All submissions for a specified item of coursework can be run against a number of sets of data. The output from the students' programs is compared with the expected output for each set of data. Time and space limits are placed on the execution of a program so as to prevent a looping program from continuing unchecked, and other steps are taken to minimise the potential for a program to damage the system.
- Submissions and the results of the testing process can be inspected on-line by authorised staff. Anonymity is preserved by storing data by University ID number.
- Students can test their programs by running them against one data set on which they will eventually be tested, and under precisely the same conditions. Thus a student can check that their program will run correctly under the final testing environment. This ensures that the program will work as the student expects when being tested and marked. In addition, it provides students with confidence that their submitted work does pass some minimal requirement.
- Final marks are stored in a SQL database and correlated with information from the University database (names and courses versus ID numbers and course registration, for example) to produce final marksheets for examination secretaries.

The BOSS system is a tool to allow students to submit assignments, and for those programs to be tested automatically. It is not an automated marking system. It is the responsibility of the individual lecturer to provide a marking scheme which takes account of the results produced by BOSS, together with all other factors which may be regarded as important (such as program style, commenting, etc).

### Resource Implications

The BOSS system has provided us with a number of benefits without compromising the general approach taken of maximising exposure to standard tools and utilities. Large numbers of students have been handled efficiently by the system, with security of assignment submission being assured. Programs submitted cannot be copied by other students, and the possibility of paper submissions being accidentally lost is removed. Secretarial staff do not need to be employed at deadlines to collect assignments, making more efficient use of secretarial time, and the volume of paperwork involved can be reduced to (almost) zero both for the lecturer and for administrative and secretarial staff.

More importantly, perhaps, the time needed to mark an assignment is reduced considerably, while the accuracy of marking, and consequently the confidence enjoyed by the students in the marking process, is improved. In addition, consistency is improved, especially if more than one person is involved in the marking process.

### Student Assessment

In this last year, we have also introduced the system into a second-year course which covers the practical application of software tools. Though this normally requires a slightly more involved testing regime, the BOSS toolkit provides a very adequate and appropriate means of automatic submission and testing. More importantly, perhaps, students have had virtually no difficulty in using the system. This seems to imply that the newly established culture has taken root, and that our initial efforts at integrating the system into the fabric of the degree courses are paying off.

### Evaluation

As it stands, the system is functioning well. There has been a generally favourable student response, and this has improved as the culture of automatic submission has become established within the Department. In addition, lecturers and tutors have also found the system to be simple and easy to use, and marking times have been reduced significantly with a corresponding increase in consistency throughout.

We sought feedback from students by means of questionnaires which required students to comment on their experiences of using the system, and also questionnaires which required numerical responses for questions relating to system use. These were generally favourable, and most students considered it an easy system to use. The ability to use the utility to test programs in advance of submission to check the conformance of their programs to the specification was also widely appreciated.

The principal concerns expressed fell into two categories. The first of these covered minor criticisms about the user-interface and the specific messages that the system provides to students when a program fails the test utility. Many of these criticisms have since been addressed in the latest version of the BOSS system, and we are continuing development so that the user-interface is improved still further.

The second – and perhaps more interesting – category of criticism was that the output expected was too precisely specified. BOSS is far too "fussy". This criticism relates to the format of the output specified – as in the precise layout of tabular output, for example – and also to some students' desire to design their own user-interfaces by establishing interactive prompting for input, for example. This is an important point, for it seemed to reflect the preference of first year undergraduates who had had considerable programming experience prior to joining our course. Many of them were thus used to programming in an unstructured fashion and were unused to being required to follow precise specifications.

### *Key Advice*

- Don't underestimate the value of a model solution to the construction of an unambiguous and precise problem specification.
- Use assessment tools for formative assessment when rapid feedback is required.

### *References*

Finkelstein, A. (1993) 'European computing curricula: a guide and comparative analysis', The Computer Journal, 36(4):299–319.

Joy, M. and Luck, M. (1995) 'On-line submission and testing of programming assignments', in J. Hart, editor, Innovations in Computing Teaching, SEDA, London.

Joy, M. and Luck, M. (1996) 'Software standards in undergraduate computing courses', Journal of Computer Assisted Learning, 12:103–113.

Luck, M. and Joy, M. (1995) 'Automatic submission in an evolutionary approach to computer science teaching', Computers and Education, 25(3):105–111.

# CASE STUDY EIGHT
# Experiences of assessing programming assignments by computer

Roger Oliver

# Experiences of assessing programming assignments by computer


### Author and contact:

Roger Oliver
Department of Computer Science
University of Hertfordshire
Hatfield
Hertfordshire AL10 9AB
UK

### Email:

R.G. Oliver@herts.ac.uk



### Key-words:

automatic assessment, computer programming, collusion, motivation



### Outline:

Experience of the use of a system to mark student programming assignments by inspection of their output and source code is described. The benefits of increased student motivation and some issues of program quality are highlighted. These can be accomplished without unreasonable staffing demands. Problems including some students failing to accept the validity of the assessment criteria, obsessive behaviour, and collusion are also reported. Suggestions are made on how some of these problems may be reduced.


### Context

This case study is concerned with how one (computer programming) course was partially assessed by computer. It does not concern itself with the technical details of computer-based assessment systems (Benford 1994, Oliver 1996), but with the educational and course management issues accompanying the use of such techniques. The case study relates to a "combined studies" course in programming. Typically, there were 45 students on the course.

### Description

The assessment of the course consisted of three elements: a series of small exercises set through the course making up 15% of the assessment, a series of class tests contributing to 60% of the assessment, and a group practical comprising 25% of the assessment.

The small exercises were marked by computer in an attempt to save teaching-staff time. They were assessed on the basis of the output produced by the program, and by inspection of the source code itself for typographical layout and for the presence of particular features; these included the presence of comments and the use of specific language constructs mandated by the exercise (e.g. use of "for" loop for a particular exercise).

A program (PEST - Program Exercise Solution Tester) was written to assess the program exercises. For each exercise a series of (Awk) marking scripts was required plus a script to inspect the source code. Plagiarism-detecting software was also used.

### Resource Implications

The demand for machine-time on the central Unix servers was very modest. The analysis of a typical exercise for the whole class took about five minutes in real-time. One of the plagiarism tests was somewhat more demanding, typically taking about 15 minutes to complete.

Once experience had been gained in writing the necessary marking scripts (Oliver 1997), it was found to take between 30 and 60 minutes to create the necessary files for a new exercise.

### Student Assessment

Computer programming is a practical skill that can only be learnt by practice. Assessing the small exercises set during the course by computer increased the motivation of the students to do them. The evidence for this was a rise in the number of errors in the exercise specifications reported by students, many more students asked questions about the details of the exercises, and more students complained about the amount of work that the course required! Sometimes, students were over-motivated (see below).

A subsidiary benefit was that the exercises often initiated a dialogue between staff and students on various issues relating to program quality. Without computer assistance, it would not have been possible to mark enough work to achieve this result.

### Evaluation

The approach adopted helped the course team more nearly to achieve the aims of the course within the context of limited resources. However, certain problems were experienced.

Firstly, some students had difficulty in accepting the validity of the assessment criteria. For example, they did not accept the need to adhere to source code style rules, and a deduction of 0.1 of a mark out of the total exercise mark of 0.5 because of poor program layout was seen as an unreasonable 20% penalty. They saw themselves completing a difficult exercise and then not receiving full credit for it.

A second problem was that staff teaching other courses complained that students were neglecting their courses because of the assessed programming assignments.

Thirdly, a few students became obsessed with completing the exercises, despite the small contribution each exercise made to the overall course assessment. The student perspective often seemed to be that a mark, however small, was very significant.

Fourthly, at least a fifth of students submitted work on one or more occasions that showed an unacceptable degree of working togather. This was despite being warned that plagiarism detection software was being used.

A final problem was the difficulty of communicating to the students what was important about an exercise and what was not. Students often failed to distinguish between design issues which had not (yet) been addressed by the course, and which therefore did not figure in the assessment, and those which had been addressed and which were assessed. Some students became frustrated by what they saw as arbitrary and opaque assessment rules.

### Key Advice

- It is far easier to design automated assessments that are intolerant of student mistakes than those that are not. Computers do not generally exhibit intelligent behavior and you should decide what will be acceptable to your students and balance that against the amount of time you can spend devising the assessments. Do not confuse what is expedient with what is educationally defensible.

- Keep your colleagues teaching other courses to your students informed about what you are doing. If you are successful in motivating your students they may neglect their other courses!
- If you set your students many exercises that put them under pressure, plagiarism and collusion will occur. Keep assessment loads reasonable!
- Distinguish between what you can assess and what you should assess at any particular point in the course. If your students are not technically or emotionally "mature" enough to accept the validity of your judgements, you will simply antagonise them.

### References

Benford, S.D., Burke, E.K., Foxley, E., Gutteridge, N.H. and Gibbon C.A. 1994 Observations of the impact of the Ceilidh System on the teaching of computer programming. In: 2nd All-Ireland Conference on the Teaching of Computing. Dublin: CTI/CTC, 29-34

Oliver, R.G. 1996 Automatic assessment of programming assigments by inspection of their output using PEST and awk. In: 4th Annual conference on the teaching of computing. Dublin: CTI/CTC, 51-55.

Oliver, R.G. 1997 Using Awk for assessment of programming assigments by inspection of their output: problems and solutions. In: 5th Annual conference on the teaching of computing. Dublin: CTI/CTC,183-186.

# CASE STUDY NINE
# Computer-based assessment in palaeontology

C. R. C. Paul and A. P. Boyle

# Computer-based assessment in palaeontology


### Author and contact:

C. R. C. Paul and A. P. Boyle

Department of Earth Sciences,

University of Liverpool,

L69 3BX

UK

### Email:

crcp@liv.ac.uk, apboyle@liv.ac.uk



### Key-words:

Palaeontology, learning outcomes, learning strategy, course organisation


### Outline:

Courseware for computer-based learning (CBL) and computer-based assessment (CBA) have been used for 3 years to replace lectures and 'open book' question papers (questionnaires) in practicals in a 2nd year palaeontology module (Boyle et al. 1997). Most students like them and believe they learn as well or better from IT materials as from lectures and practicals, but this perception is not always supported by exam results. Students complete CBAs in much less time than the paper questionnaires. The student procedure is to attempt a CBA as a pre-assessment of the subject before using the relevant courseware, and to repeat the CBA if they gain <50% initially. However, many repeat the CBAs to learn from them. When enough courseware and CBAs are available, the module will be reorganised around concurrent projects rather than its current linear structure, since there is no reason why students should learn about fossils in any particular order. This will make much better use of limited practical resources.

### Context

The example involves a second year palaeontology module in Honours B.Sc. degrees in Geology and Geology & Physical Geography. Classes average 35 students. The module runs for 6 weeks with one hour lectures run twice and one 3-hour practical each week. It covers five fossil groups: corals, cephalopods, gastropods, bivalves and echinoids.

### Description

Over the last three years we have been introducing CBL and CBA to replace lectures and paper questionnaires for some of the five fossil groups covered, thus enabling comparison of learning outcomes between conventional and IT-delivered teaching. Paper-based questionnaires were originally intended to reinforce lectures and facilitate learning, so students are encouraged to use lecture notes, textbooks, group discussions, etc. to reach answers. Practical material is available for one week, since students rarely complete practical work within the allotted time. All questionnaires are marked, and returned with comments by the next practical.

CBAs were introduced to replace questionnaires. Both generally test knowledge, but some questions are intended to assess reasoning too. CBAs are in 6 sections which progress from "name this part" style to higher-level questions such as "which of the following reasons supports your previous interpretation?" These CBAs have substantial built-in redundancy by using random selection techniques. The aim is to test each student equally thoroughly, but to generate a different assessment every time a student logs on. For example, in the morphology section the computer randomly selects five terms from about 12, and one photo from 5 or 6, for each selection. Students cannot exhaust all combinations. Thus, although the most common reason given for repeating CBAs is to improve marks, this does not always happen. Typically students may improve their scores over the first four attempts but thereafter reach a plateau. The best mark is recorded to avoid students repeating CBAs endlessly to equal some earlier score.

Student feedback forms have been used to assess their reactions to IT materials. In addition, part of the summative traditional final exam includes 5 short questions on each fossil group.Thus it has been possible to compare actual learning outcomes on basic knowledge from conventional and IT-delivered learning. In particular, two-thirds of students thought they learnt as well or better from IT materials as from conventional teaching. Exam results (Table 1) do not always confirm this perception; although statistically significant differences occur between the two methods they are not consistent.

### *Resource Implications*

Courseware and CBAs are very expensive to create in terms of hardware, software and staff time. However, once available in subjects such as basic palaeontology, their shelf life is very long. When we have IT materials that work, we put them on the network and leave them there. Only when all groups (of both macro- and micro-fossils) have been covered will we upgrade existing materials, apart from correction of bugs or factual errors. Savings in staff time so far amount to about 3 person-days per year (for a module with 30 contact hours per year). Greater time savings occur for students. They complete CBAs in 25 minutes on average: in only two cases have students taken more than 1 hour. Many complete later repeats in 5 minutes. Paper questionnaires are rarely handed in by the end of a practical, suggesting that most students take >2.5 hours to complete them. Furthermore, data on time taken versus marks gained mean that students can be advised on learning strategies. Their best strategy is to attempt a CBA first, then if they do badly to review the courseware; if not they should attempt the next CBA. Only if they have significantly less than 50% is it worth repeating a CBA, in terms of marks per minute. Nevertheless, many students do repeat CBAs. The second most common reason given is to learn from them.

### *Student Assessment*

This is summarised in the tables. Courseware and CBAs have been added gradually, so there are different comparisons between IT and conventional teaching each year. The null hypothesis is that there is no difference between mean scores for different groups. This can be rejected at the 95% confidence level if values are <0.05 (green values in tables 1-3). So far performance seems to relate to fossil group as much as to method of teaching: echinoids are consistently done well, cephalopods fluctuate between best and worst results despite no change in teaching method. Bivalves are consistent despite changing the teaching method this year.

### *Evaluation*

The strengths of the IT materials are that students enjoy them (perhaps the novelty factor). They like the flexibility and privacy (they don't make mistakes in public), but learn from the instantaneous feedback. So far, at Liverpool, hardware resources have expanded faster than demand (at least in palaeontology teaching), but still represent a major investment. Our IT materials are always available on the university network, giving excellent flexibility in time and place of study. So far, we have no objective data on learning times, but student feedback suggests they spend as long studying courseware as they would have attending lectures. The major drawback is the time it takes to produce good quality CBA with feedback. Educational advantages none the less make this worthwhile. Students do not need to learn about one fossil group before another one. The main problem with conventional large classes is providing adequate practical materials when the entire class is studying the same fossil group. Once enough IT materials exist (hopefully in the 98-99 academic year), the module will be rearranged around projects. Students will select a project, and then review the appropriate courseware and CBA first. They will study general principles and applications intensively for one group via the project, and complete weekly courseware and CBAs on other groups just to cover basics. All projects will run concurrently for the entire six weeks, utilising scarce practical resources more efficiently.

### *Key Advice*

With respect to CBAs, it is helpful to consider the following suggestions:

- Start by defining the learning outcomes and assessment strategy. Which outcomes will be assessed by which methods?
- Decide how CBA fits into the strategy.
- Decide if CBA will be formative or summative. How will feedback be provided? Note that there is little point in creating CBAs that do not provide feedback. The greatest strength of CBAs is that students can learn from them, especially if the feedback is customised to the individual student and is diagnostic and ranged.
- Assemble all materials and plan (storyboard) the layout before you start. If you can, develop CBAs with someone not expert in the subject matter. If the questions are clear to them, they are more likely to be clear to students.
- Start with a simple example (e.g. computerise a quiz you already have). Try it out before using it for summative assessments.
- Have a conventional fallback position in case the network crashes or the students discover a serious bug you have overlooked. Don't forget you know what is intended, but students have an unbelievable knack of doing the unexpected!

**Table 1. Summary of 1996 t-statistics** (lower part of table) and probabilities (upper part of table) for paired sample t-test with Ho: $\mu=0$ Ha: $\mu\neq0$ for a = 0.05 and 53 d.f. Probabilities less than 0.05 (green) indicate rejection of null hypothesis. Titles in italics were taught and assessed by CBA and CBL materials. [1]Cephalopods were not assessed by any means in 1996 because the CBA was not ready in time.

| Group | Corals | [1]Cephalopods | Gastropods | Bivalves | Echinoids |
|---|---|---|---|---|---|
| Corals | - | **0.0026** | 0.1025 | 0.7564 | **0.0201** |
| Cephalopods | 3.163 | - | 0.2003 | **0.0014** | **0.0000** |
| Gastropods | 1.662 | 1.297 | - | 0.0659 | **0.0001** |
| Bivalves | 0.312 | 3.380 | 1.878 | - | **0.0024** |
| Echinoids | -2.397 | 5.002 | 4.185 | 3.184 | - |

**Table 2. Summary of 1997 t-statistics** (lower part of table) and probabilities (upper part of table) for paired sample t-test with Ho: μ=0 Ha: μ≠0 for a = 0.05 and 37 d.f. Probabilities less than 0.05 (green) indicate rejection of null hypothesis.

| Group | Corals | Cephalopods | Gastropods | Bivalves | Echinoids |
|---|---|---|---|---|---|
| Corals | - | **0.0004** | 0.5236 | 0.9645 | 0.0789 |
| Cephalopods | -3.910 | - | **0.0001** | **0.0001** | **0.0166** |
| Gastropods | -0.644 | -4.521 | - | 0.0499 | 0.3033 |
| Bivalves | -0.045 | 4.353 | 0.568 | - | 0.1783 |
| Echinoids | -1.807 | 2.511 | -1.044 | -1.372 | - |

**Table 3. Summary of 1998 t-statistics** (lower part of table) and probabilities (upper part of table) for paired sample t-test with Ho: μ=0 Ha: μ≠0 for a = 0.05 and 33 d.f. Probabilities less than 0.05 (green) indicate rejection of null hypothesis.

| Group | Corals | Cephalopods | Gastropods | Bivalves | Echinoids |
|---|---|---|---|---|---|
| Corals | - | **0.0000** | **0.0001** | **0.0347** | 0.8435 |
| Cephalopods | 5.776 | - | 0.4349 | **0.0002** | **0.0000** |
| Gastropods | 4.676 | 0.795 | - | **0.0499** | **0.0002** |
| Bivalves | 2.252 | 4.540 | 2.075 | - | 0.0530 |
| Echinoids | 0.200 | -5.500 | -4.460 | -2.045 | - |

**Table 4. Summary of means and standard deviations for all five fossil groups in years 1996, 1997 & 1998.**

| Cohort | No. | Corals | | Cephalopods | | Gastropods | | Bivalves | | Echinoids | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | x | s.d. | x | s.d. | x | s.d. | x | s.d. | x | s.d. |
| 1996 | 54 | 3.43 | 1.11 | 2.81 | 1.21 | 3.07 | 1.23 | 3.37 | 0.92 | 3.91 | 0.98 |
| 1997 | 38 | 3.40 | 1.20 | 4.45 | 1.08 | 3.55 | 0.80 | 3.41 | 1.41 | 3.79 | 1.23 |
| 1998 | 23 | 3.43 | 0.95 | 2.41 | 2.41 | 2.67 | 1.28 | 3.37 | 1.00 | 3.93 | 1.09 |

## *References*

Boyle, A.P., Bryon, D.N. & Paul, C.R.C. 1997. Computer-based learning and assessment: a palaeontological case study with outcomes and implications. Computers and Geosciences, 23: 573-580.

# CASE STUDY TEN
# Summative assessment on the WWW for a credit bearing evening class

Dave Whittington

# Summative Assessment on the WWW for a Credit Bearing Evening Class


### Author and contact:

Dave Whittington

Department of Computer Science

University of Strathclyde

26 Richmond St.

Glasgow

G1 1XH

UK

### Email:

dave@cs.strath.ac.uk


### WWW URLs:

CVU Assessment Hall - http://cvu.strath.ac.uk/assessment/

CVU Assessment Engine - http://cvu.strath.ac.uk/ae/


### Key-words:

Summative assessment, virtual universities, WWW, evening lasses


### Outline:


This case study describes how Web technology has been used to develop a virtual university infrastructure and how that infrastructure has provided the platform for delivering summative assessments to a credit bearing evening class.


### Context

Clyde Virtual University (CVU), which is funded by the Scottish Higher Education Funding Council through its "Use of the MANs Initiative", is dedicated to the delivery of Internet-based multimedia learning materials to students at institutions connected to the ClydeNet MAN. Led by Strathclyde University, CVU is a collaborative project involving four additional participating institutions: Glasgow University, Glasgow Caledonian University, Paisley University and Glasgow School of Art.

As well as presenting tutorial material in an innovative and engaging format, CVU has enhanced the interactivity of these modules by adding a variety of online self assessment tests providing students with ongoing feedback on their learning experience. More detailed self assessment tests can be found in the CVU Assessment Hall (http://cvu.strath.ac.uk/assessment/). At present, these more comprehensive assessments, designed to compliment the tutorial modules in the Lecture Theatre, cover Aerospace Engineering, Organisational Behaviour and Information Technology.

The assessment mechanism developed in order to deliver self assessments can also be used to deliver summative assessments.

Since 1994 the University of Strathclyde's Continuing Education Department has held "Exploring The Internet" evening classes. These classes provide an opportunity for adults, who are not registered students, to come in to the University, learn about the Internet and the World-Wide Web and make use of the University's computing and network resources. In 1996 it was decided that the class, which had previously run as a "Leisure" class, should become "credit bearing". Having some form of student assessment was part of making the class credit bearing. As the class is all about the Internet, it made sense to assess it using the Internet.

### *Description*

The class, Exploring the Internet, is open to all and covers the World-Wide Web, email and newsgroups. There are four assessments which the students must complete: Browsing, Searching, Email and Newsgroups. These assessments match the way the course content is delivered. The assessments are presented to the students as Web pages with ten mult-ichoice and multi-response questions on each page (using HTML form elements). After the students have selected what they think are the correct answers, they click on a button at the bottom of the page which sends their answers back to the assessment mechanism where they are marked. Feedback is given to students immediately detailing the right and wrong answers and an email detailing the student's attempt is sent to the class tutors. The students use the feedback they are given to decide where further study (if any) is required. The students may take the assessments as many times as they wish, the aim being to ensure that they have understood all the material and not to fail those who did not immediately understand it. The email sent to the tutor contains login information (user-id, date and time and machine name) so that the tutor can easily identify whose attempt has generated which email message. Perhaps because this is an adult education class and a tutor is present at all times, cheating has never been a problem. Students who complete the four assessments and have a satisfactory attendance record are awarded three SCOTCAT points.

### *Resource Implications*

As with most CBA activity, the main resource burden is split between the development and the delivery of the tests. Using the Web simplifies the delivery of the tests: in this case the students were provided with Web access and training in its use as part of their class. Marking the tests is fully automated and takes up no time.

### *Student Assessment*

Not all students who take the evening class complete the assessments. The class began life as a leisure class which was taken for fun or to satisfy curiosity; not all the students attending the class are interested in the option of gaining three SCOTCAT points. Other students take the class repeatedly just to have tutored Internet access and will have already taken the assessments and been awarded the SCOTCAT points. However, all the students who want the credit manage to pass the tests.

### *Evaluation*

Around 50 students have so far taken the new accredited evening class and completed the assessments online. Some of them will not have done any form of assessment for a long time but they all find it straightforward.

### *Key Advice*

- The mode of assessment should not be dissimilar from the mode of learning
- Students should be familiar with the technology BEFORE the assessment
- Be prepared for things to go wrong: have “fallback” arrangements available

# CASE STUDY ELEVEN
# The use of computerised assessment in health science modules

Lynne Wybrew

# The use of computerised assessment in Health Science modules


### Author and contact:

Lynne Wybrew
Dept. Health Science & Biology
University of Luton
Park Square
Luton
Bedfordshire
LU1 3JU
UK

### Email:

lynne.wybrew@luton.ac.uk



### Key-words:

formative and summative computer-based assessment, anatomy & physiology, cell biology, level one



### Outline:

Computer-based assessment, using 'Question Mark' software, has been adopted for level one modules in Human Anatomy & Physiology (A&P), and Foundation Cell Biology (FBS) at this University since 1996. Prior to this, the students were assessed continually by optically mark read multiple choice questions (MCQ). The introduction of end-of-module computer-based assessment has had no adverse affect on student performance, reduced lecturer marking time but at the expense of no longer providing continual feedback for the student. This has been partially addressed by the use of formative continual assessment, but requires further investigation.


### Context

Within the fields of Life Sciences, the two A&P modules are core for single/major/joint/minor awards in Health Science, Pharmacology, Human Biology, Sport & Fitness Science and Sport & Fitness Studies. They are option modules in the major/joint/minor awards of Human Biochemistry and the minor award of Immunology. They are also core modules for HND Health Science and Sport Science and an option for Biology. Student numbers range from 350 - 400 per module (run in separate semesters). Both of these modules provide a comprehensive introduction to A&P and are taught through student-centred methods (short directed lectures and tutorials) and aim: -

- to provide a broad knowledge of human anatomy & physiology
- to relate structure to function
- to correlate the interaction of different body systems
- to encourage the students to become independent learners and promote the use of transferable skills

FBS was designed for students with little or no scientific background and as such is an option for most life science awards and a core for HND Sport Science. Student numbers range from 150 - 300 depending on student recruitment. The module explores the structure of atoms, molecules and the nature of chemical bonds; and then goes on to investigate the fundamental properties of macromolecules and how these relate to biological systems.

It introduces students to the organelles of a cell and aims: -

- to provide a basic outline of the chemistry related to life
- to introduce basic biochemical processes and concepts
- to introduce sub cellular organisation and the variety of cell types
- to relate cellular structure to function
- to provide an introduction to cell division and the basis of heredity

### Description

In all of these modules the students have two points of assessment. The first may involve the interpretation of physiological data presented as a practical report (A&P I), consist of an open book in-course test based on two case histories (A&P II) or be the combination of one short in-course test and a worksheet (FBS). The second part of the assessment in the end-of-module exam for all three modules, is an objective test using 'Question Mark' software. Each exam contains between 60 and 80 questions and lasts between an hour and an hour and a half. Computerised self-assessment banks of questions are available for A&P students during the teaching of both modules. The questions are drawn from libraries and change frequently to match the current topic being covered. The students have open access to these self-assessment tests through the learning resources centre. A small (5 to 10 questions) sample self-assessment tests is provided for FBS students through the same outlet a few weeks prior to the exam to allow them to familiarise themselves with the software and the question types.

### Resource Implications

The University's computerised assessment system is supported by a central support unit, the Unit for Learning Technology, Research and Assessment. This unit is responsible for: supporting academic staff in question and test design; compiling and mounting tests and examinations on the university network; and liaising with Computing Services and the Examinations Office. Academic staff are encouraged to concentrate on writing good questions and considering their overall assessment strategies, rather than on learning to operate the software or handling the operational issues of computer-based assessment. This means that savings in terms of staff time are maximised.

The University has an IT Suite with approximately 180 PCs which are used to run end-of-module exams during the exam weeks at the end of each semester. While this alleviates problems concerning large groups and timetabling, other issues such as the overall management of IT resources during this period become important. Any invigilated formative tests which take place during the module must be conducted in Faculty computing areas. These do not have as many PCs as the central resource and running such tests presents problems of accommodating large groups, competing for timetabling with teaching classes, and the provision of technical support

### Student Performance

There are no "before and after" student performance records available for comparison in the FBS module as it has been assessed using computer-based assessment from the outset. The module was the amalgamation of two half modules (Basic Chemistry and Foundations of Cell Biology). In the A & P modules comparison between student results before and after adopting computer-based assessment reveals that there is no difference in the average mark or in the percentage pass rates for the modules.

### Evaluation

The end-of-module exams covers all aspects of the module and the questions relate to the learning objectives given to the students for each topic. The questions which deal with physiology try to avoid ambiguity by using independent statements so that each item can be considered separately. The design of questions requires repeated use and subsequent analysis of the answers to avoid the risk of ambiguity. This allows questions to be developed which require logical deduction. The evaluative process involves feedback from students and the statistical analysis provided by the software.

Matching questions, while very good for factual recall, are also used successfully to relate structure to function in basic cell biology. However, the use of diagrams for labelling has not been successful, mainly due to insufficient clarity in the diagrams chosen. Short answer questions have not been attempted due to the variety of spelling alternatives.

Once a bank of 'tried and tested' questions is set up, the exam preparation is much quicker and easier for the lecturer - and the computerised marking of 400 tests is a major blessing. The length of time required in actually taking the exam for the students is decreased and the opportunities for self-assessment reduce anxiety. Self-assessment also increases the students' independence (in learning) and gives them confidence in accessing the computers. However, it does require repeated reminders that the facility is available to them.

Prior to the computer-based assessment, the students were continually assessed (every 2-3 weeks) using compulsory optically mark read tests. While this was time-consuming for the lecturer, it provided the students with instant feedback and a gauge of 'how well they were doing'. Using the same format with computer-based testing is a resource issue and remains the biggest weakness of our approach. While self-assessment banks can provide feedback there is no guarantee that students will use them.

In the FBS module, we are in the process of developing a computerised teaching package which will allow students to access the lecture material and related assessment questions. However, this process will need to remain formative until the resourcing issue is addressed.

### Key Advice

**Do: -**

- Have prior experience with the question type, preferably used formatively with a group of 'vocal' students.
- Provide feedback in the formative tests for each question - this can be done primarily by telling the students which answe is right or wrong. The next step is to have a feedback that explains why they were right or wrong and gives references to text books.

**Don't: -**

- Assume the student is computer literate
- Assume the computer resources will be available - have a paper back up for emergencies

## Other specific examples of CBA in science and computing:

There are a very large number of examples of CBA in use in the UK and elsewhere. Listed below is a small selection of recently published case studies.

Beevers, C.E., McGuire, G.R., Stirling, G. & Wild, D.G. (1995) Mathematical ability assessed by computer. Computers and Education, 25, 123-132.

Lloyd, A. & Holden, C. (1994) Computer aided assessments in pharmaceutical microbiology. Active Learning, 1, 27-28.

Martin, J.G., McCaffery, K. & Lloyd, D. (1996) The introduction of computer-based testing on an engineering technology course. Assessment and Evaluation in Higher Education, 21, 83-90.

Parrington, N., Ferguson, I., Hedges, S. & Spence, L. (1994) The use of hypertext and the World Wide Web in teaching and assessing software engineering. Active Learning, 1, 39-42.

Proctor, A. & Donohue, D. (1994) Computer based assessment: a case study in geography. Active Learning, 1, 29-34.

Sensi, S., Merlitti, D., Murri, R., Palitti, V.P. & Guagnano, M.T. (1995). Evaluation of learning progress in internal medicine using computer-aided clinical case simulation. Medical Teacher, 17(3), 321-326.

Thoennessen, M. & Harrison, M.J. (1996) Computer-assisted assessment in a large physics class. Computers and Education, 27(2), 141-147.

# Further Information

### *Subject CTI centres*

The Computers in Teaching Initiative operates through subject based centres who offer a variety of support in IT in teaching. Activities vary between CTI centres but all will have useful information and resources relating to CBA in their individual science and computing subjects. There are 24 centres in all but only those most relevant to science and computing are detailed here. Contact the CTI Support Service for further details on other centres. The information below is accurate at the time of going to press, but please bear in mind that there is likely to be some re-organisation of the CTI network in the summer of 1999 or soon afterwards. Again, the CTI Support Service will be able to advise on this.

**CTI Support Service**
University of Oxford
13 Banbury Road
Oxford
OX2 6NN
Tel: 01865 273273
Fax: 01865 273275
Email: ctiss@oucs.ox.ac.uk
http://www.cti.ac.uk/

**CTI Biology**
Donnan Laboratories
University of Liverpool
Crown Street
Liverpool L69 7ZD
Tel: 0151 794 5118
Fax: 0151 794 4401
Email: ctibiol@liverpool.ac.uk
http://www.liv.ac.uk/ctbiol.html

**CTI Centre for Chemistry**
Donnan Laboratories
University of Liverpool
Crown Street
Liverpool L69 7ZD
Tel: 0151 794 3576
Fax: 0151 794 3586
Email: ctichem@liverpool.ac.uk
http://www.liv.ac.uk/ctichem.html

**CTI Computing**
Faculty of Informatics
University of Ulster at Jordanstown
Newtownabbey
Co Antrim
BT37 0QB
Tel: 01232 368020
Fax: 01232 368206
Email: cticomp@ulst.ac.uk
http://www.ulst.ac.uk/misc/cticomp/

**CTI Geography, Geology & Meteorology**
University of Leicester
Leicester
LE1 7RH
Tel: 0116 252 3827
Fax: 0116 252 3854
Email: cti@le.ac.uk
http://www.le.ac.uk/cti/

**CTI Centre for Land Use and Environmental Sciences (CTI-CLUES)**
MacRobert Building
University of Aberdeen
Aberdeen
AB24 5UA
Tel: 01224 273754
Fax: 01224 273752
Email: cticlues@aberdeen.ac.uk
http://www.clues.abdn..ac.uk:8080/

**CTI Centre for Medicine**
University of Bristol
Institute for Learning and Research Technology
8 Woodland Road
Bristol
BS8 1TN
Tel: 0117 928 7492
Fax: 0117 928 8473
Email: cticm@bristol.ac.uk
http://www.ilrt.bristol.ac.uk/cticm

**CTI Physics**
Department of Physics
University of Surrey
Guildford
GU2 5XH
Tel: 01483 259329
Fax: 01483 259401
Email: ctiphys@surrey.ac.uk
http://www.ph.surrey.ac.uk/cti/

### *Other organisations:*

**Teaching and Learning Technology Programme (TLTP*)*** has provided the stimulus for a variety of technology based learning materials across all subjects. Materials are available at cost to HE institutions from the individual projects, their publishers or centrally from TLTP.

**TLTP**
Northavon House
Coldharbour Lane
Bristol
BS16 1QD
Tel: 0117 931 7454
Fax: 0117 931 7173
Email: tltp@hefce.ac.uk
http://www.tltp.ac.uk/tltp/

**Association for Learning Technology (ALT)**

ALT promotes good practice in the development of learning technologies in HE. Its aims are to:

- promote good practice in the use and development of learning technologies in higher and further education
- facilitate interchange between practitioners, developers, researchers and policy makers in education and industry
- represent the membership in areas of policy such as infrastructure provision and resource allocation

Information on publications etc. is available on-line but members receive copies of these and other privileges. Their web site contains a lot of very useful information and can be found at http://www.warwick.ac.uk/alt-E/

*Notes*

*Notes*

*Notes*

## THE SEED PROJECTS AND THEIR CONTACT DETAILS

***Project 1: A web based bibliographic database on Science teaching and learning, designed to support the information requirements of the SEED Projects.***
Nigel May, Science Faculty Team Co-ordinator,
Tel: 01752 - 232318, E-mail: nmay@plymouth.ac.uk.

***Project 2: An investigation of the potential development of Curriculum Support Teams.***
Nigel May, Science Faculty Team Co-ordinator,
Tel: 01752 - 232318, E-mail: nmay@plymouth.ac.uk.

***Project 3a: A handbook on field teaching in the Sciences.***
Colin Williams, Geological Sciences,
Tel: 01752 - 233103, E-mail: c1williams@plymouth.ac.uk.

***Project 3b: Field discovery days.***
Colin Williams et al, Geological Sciences,
Tel: 01752 - 233103, E-mail: c1williams@plymouth.ac.uk.

***Project 4: Fieldwork issues and developments.***
Les Ternan, Geographical Sciences,
Tel: 01752 - 233060, E-mail: jternan@plymouth.ac.uk
and Andy Elmes, SEED Programme,
Tel: 01752 - 233532, E-mail: aelmes@plymouth.ac.uk.

***Project 5: A handbook on laboratory teaching.***
Les Jervis, Biological Sciences,
Tel: 01752 - 232929, E-mail: ljervis@plymouth.ac.uk.

***Project 6: Peer assisted learning strategies (Supplemental Instruction) (P.A.L.S (S.I.).***
Stuart Johnston, Educational Development Services,
Tel: 01752 - 233317.

***Project 7: Development of a framework for the training and management of graduate teaching assistants.***
Rhona Sharpe, Educational Development Services,
Tel: 01752 - 232346, E-mail: rsharpe@plymouth.ac.uk.

***Project 8: Development of a computer-aided learning package for environmental organic chemistry.***
Steve Rowland, Environmental Sciences.
Tel: 01752 - 233013, E-mail: srowland@plymouth.ac.uk.

***Project 9: Environmental issues in the Mediterranean: a case study of the Maltese Islands.***
John Stainfield, Geographical Sciences,
Tel: 01752 - 233069 - E-mail: jstainfield@plymouth.ac.uk.

***Project 10: Computer based assessment in science: a review of good practice.***
Dan Charman, Geographical Sciences,
Tel: 01752 - 233058, E-mail: dcharman@plymouth.ac.uk
and Andy Elmes, SEED Programme,
Tel: 01752 - 233532, E-mail: aelmes@plymouth.ac.uk.

***Project 11: CAL and basic Science.***
Neil Witt, Institute of Marine Studies,
Tel: 01752 - 232417, E-mail: nwitt@plymouth.ac.uk.

***Project 12: A handbook on employer-links in Science.***
Stuart Lane, Biological Sciences, Tel: 01752 - 232908,
E-mail: slane@plymouth.ac.uk and Mandy Burns,
Learning and Research Support Services,
Tel: 01752 - 232255, E-mail: mburns@plymouth.ac.uk.

***Project 13: Using multimedia for providing feedback to students undertaking concurrent project-based practicals.***
Graham Bradley, Biological Sciences,
Tel: 01752 232934, E-mail: gbradley@plymouth.ac.uk and
David Gaudie, Biological Sciences,
Tel: 01752 - 232945, E-mail: dgaudie@plymouth.ac.uk.

***Project 14: An environmental data base for projects in environmental impact assessment (EIA) and conservation.***
Andrew Williams, Geographical Sciences,
Tel: 01752 - 233059, E-mail: awilliams@plymouth.ac.uk.

***Project 15: Webkit - a toolkit to produce interactive web pages in support of CAL.***
Kevin Rowley, School of Computing,
Tel: 01752 - 232621, E-mail: krowley@plymouth.ac.uk.

***Project 16: Qualifications update in applied Science for industry.***
Mike Lister, Cornwall College,
Tel: 01209 - 712911, E-mail: m.lister@cornwall.ac.uk.

***Project 17: Baseline assessment of competencies and skills for Science and Computing.***
Dave Croot, Geographical Sciences,
Tel: 01752 - 233070, E-mail: dcroot@plymouth.ac.uk and
Stanley Oldfield, School of Computing,
Tel: 01752 - 232552, E-mail: soldfield@plymouth.ac.uk.

## SEED aims to:

## Document...

## Develop...

## Disseminate... best practice in Science Education


### The SEED Programme

Faculty of Science
University of Plymouth
Drake Circus
Plymouth PL4 8AA

Tel: 01752 233530
Fax: 01752 233534

Programme Manager: Brian Chalkley
e-mail: bchalkley@plymouth.ac.uk

Programme Officer: Andrew Elmes
e-mail: aelmes@plymouth.ac.uk

http://www.science.plym.ac.uk/departments/seed/