Integration of Software into Course Development

Simone Richardson
CTI Centre for Physics, University of Surrey, United Kingdom

This article aims at assisting educators in the production of courses of the highest quality. In particular it will promote the application of software in physics courses.

Development of courses

Developing courses often seems to not only involve reinventing the wheel but also the production of an entire garage forecourt. What one really wants to do is produce a course that builds on the feedback from previous courses, with the minimum of effort and that everyone else wishes they had produced.

As accountability becomes more prominent in teaching as well as research, it is important for a lecturer to be able to show that a course fully meets its aims and objectives. It is very tempting for more experienced staff to ignore such criteria, but nevertheless necessary when an evaluation of the impact and effectiveness of the course[13,14] is made. It is also then possible for someone else to run a course you have written and be confident about the overall goals. The preparation of a course can begin by working through the following list of actions:

- Look at the syllabus and produce a list of aims and objectives for the course.
- Search out what resources are available - in your present department, in your archives at home and any other sources that come to mind e.g. Institute of Physics web site PhysicsWeb (http://www.physicsweb.org/resources/).
- Review notes from previous lectures and tutorials, on similar topics. This is a good preliminary step for those new to writing courses.
- Design a scheme of work - consisting of a set number of lectures and tutorials and usually some form of assessment. At this stage assign which part of the syllabus each lecture and tutorial covers and where appropriate software could be used.
- Plan each lecture and tutorial - include which aims and objectives are to be met, prepare lecture notes and which resources (e.g. software, books) are needed. Be clear about exactly what you intend the students to learn. It is important to allow room for flexibility to cope with later ideas and unforeseen circumstances.
- Produce an appendix of any material that concerns the same topic but was not used.
- Prior to the final production of the course materials it is important to ask someone, usually another colleague, to comment constructively about the draft of the course in order to produce a good workable version.

There are a range of issues that hinder the development of a good course, but if one recognises these problems it is possible to try and reduce their effects. There is always a limit on the amount of time for planning and money available for resources, so it is important to plan carefully to be able to include all of the priorities into a course. The careful balance of staff/student contact time to non-contact time is vital to enable the students sufficient time to complete work successfully.

The access to all known sources of useful information will enable the course to be kept up-to-date and as accurate as possible. The motivation of students will affect their learning, but the situation can be improved by the lecturer being both enthusiastic and by presenting material in different ways. For example, when highlighting a particular physics concept, a lecturer could use software to introduce 3-D models of physical situations[10,14] or even using a novel presentation style (e.g. party tricks with a physics theme - pulling the cloth from under the galvanometer).

Integrating software into a lecture

A lecture is effective when it enables the students to reach their full potential. At this point we need to consider what is the purpose of a lecture to then be able to decide what the lecturer and students should be doing during it. In most courses lectures are used as the main medium for transferring knowledge to students. Most lecturers also structure their lectures in order to encourage students to think and learn, and not to just prepare for examinations with the minimum of effort. Tutorials and assignments are used to consolidate ideas already introduced and for instructing students in particular methodologies, so that they begin to be involved in deep learning.

Once lecturers have produced their lecture notes (it is a good idea to word process them) they need to decide how they will communicate with the students during each lecture session. They can display and/or read their prepared notes using an overhead projector or a computer (e.g. using *PowerPoint* and a digital projector) or distribute them on paper. But it needs to be considered that when the students are exposed to large amounts of information visually and verbally some of them will become either disinterested or confused. So to overcome this potential problem, the visual input could consist of just key points presented as static images and particular models presented using video clips to support and complement the spoken word. For instance, digital video cameras can be used to add still or moving images from experiments into lecture material. Finally, lecturers could produce a paper copy of their complete notes for distribution to students or make them available on the web with the inclusion of any available motion or still images. The presence of notes on the web provides a flexible, immediate and inexpensive resource.

The preconceptions that students arrive with strongly influence how new information is processed. Faulty preconceptions lead to confusion, even the rejection of new ideas. Therefore, it is important to consider what the student does before, during and after each lecture to know whether any deep learning will take place. During the lectures most students do something between the two extremes from just listening to writing down everything said by the lecturer, most often it takes the form of recording just the key points and diagrams. A course with a clear structure that includes suggestions of material (e.g. particular web sites, software packages, journals in the library etc.) to be reviewed prior to particular lectures focuses the students. Thus the completion of appropriate assignments using software (e.g. examination questions databases, electronic debates[16]) can be used as an alternative form of stimulus to help with the reinforcing of ideas and concepts introduced in lectures[15].

Integrating software into a course

So far it has been suggested that software (such as word processing and *PowerPoint*) can be used as a tool to support more traditional teaching methods, but it is also necessary to mention how it can become a more significant part of the course. There are a variety of software packages (e.g. *SToMP*, *Crocodile Clips* and *Interactive Physics*) that provide models of physical situations and experiments with changeable parameters which can be used for demonstrations during lectures, and as part of exercises, to assist students with grasping difficult concepts[10]. In traditional lectures a constant learning experience is maintained for all students, thus failing to take into account the variation in prior knowledge of students[5]. Whereas when using software the students can work through at their own pace which can help to accommodate their range of differing needs. For example, by use of the hypertext system on the web students can access a combination of traditional lecture notes plus supplementary material in an order that suits their needs which promotes open learning and hopefully deep learning. To ensure that students take an active part in their learning[9, 11] the students can be transferred into laboratories to take part in CAL work, in place of a lecture or a tutorial. Depending on the available time and money the tutor can also direct students to use the Internet to access courseware placed on a web server or to search for relevant materials[12]. This will not only allow students to try out exercises, but will provide a different working environment to promote interest.

Lecturers can keep their workload down by using any available facilities that will do some of the work for them. Now with the availability of CD-ROMs (e.g. *Current Contents*) and the web, they can access databases, specialised sites and journals to gain information about what software is available, what courses are available, what particular software can do and what software has been used before. This task may initially seem daunting, but there are on-line services (e.g. specialised search engines and catalogues) offered by a variety of institutions that will help. This is important so that one is able to search through all available information to find what is useful and accurate[7]. In fact it is good to encourage students to use a similar process in order to increase their knowledge source.

Another very positive reason for introducing software into courses is the way it distracts the person's attention from other aspects of the course. Some people find using a computer fun and so they can have difficult subject matter introduced to them with much more ease[9]. While others less proficient in using computers find their positive move along a learning curve includes new computer skills and the new concepts on the subject being studied. I have observed that novice computer users, particularly adults, only seem to have a problem using computers when they are afraid of damaging things and believe that an understanding of how a computer works is essential before they can actually use one. The average confidence of users will hopefully improve, as computers become an even more integral part of all our lives. This leads onto my next point concerning the ever-increasing presence of computers in leisure time activities and how it is having a noticeable effect on young people. Their attention spans are much reduced in the context of traditional delivery methods, which contain mostly verbal and written input. In contrast to the older generation, as the types of input are increased, their interest increases.

There are some points that need to be considered that do not support the idea of increasing the use of computers on courses. It has been observed that students (at the University of Surrey) enjoy CAL software so much that they are reluctant to work on material that is not based on the computer. This confirms the importance of having a clear structure for the course that includes deadlines. If the inclusion of more software leads to the gradual removal of human contact it is possible that it will become difficult for tutors to pick up on individual's problems. Finally, there is so much information available both on the web and in traditional libraries that it can become confusing and time consuming when students are trying to do some research.

Choosing the right software for your course

One easy mistake that can be made when choosing suitable software for a course is its cosmetic value. For example, in some software on the market a lot of work has gone into producing incredible graphics that astound you. But when you try and analyse its effective teaching value you will find it very limited. It could be that its material is unsuitable for the audience, that it covers too many subjects in broad terms, that it covers one subject in too much detail or unfortunately it contains inaccurate information. In general, the most suitable software to promote effective teaching is that produced by teams of people, including some team members who are still involved in teaching the particular topic covered e.g. Open University *Discovering Science* software (http://www2.open.ac.uk/ces/courses/courseFrame.html).

Before choosing software decide on the priorities of your course and then with consideration of the answers to the questions below, the most suitable software will seem more obvious.

- Finances - How much money is available? How much will the software cost to be used on the course? What percentage is this of your whole budget? Are there educational discounts? How much money is available for printed notes?
- Location and Structure of Course - In what location will the software be used (lecture, tutorial, practical or home study)? What is the structure of the course (lectures, tutorials, practicals and/or home study)? How much time is available for the use of software? What resources are available (computers, digital projector etc.)?
- Aims and Objectives of Course - What are the aims and objectives of the course? Which concepts and ideas need to be mentioned, understood vaguely, understood exactly? Which expressions, equations and formulae need to be used within exercises? What practical work needs to be done?
- Audience - What is the intended audience of the software? What are the computer skills of the students and teachers? What is the ability of the intended students?
- Available Resources - Is the software the only practical solution? What material would be used if the software were not used? Is there any alternative software that is suitable? Is the software an alternative delivery method to conventional methods (lectures) e.g. CAL? (It is usually considered to be more interesting and more flexible.)
- Software - Has anyone used this software in teaching already? Is the software up-to-date? Is the software actually necessary to teach the syllabus e.g. CAD (Computer Aided Design), FEA (Finite Element Analysis) and CAM (Computer Aided Manufacturing)? Is the software needed as part of the course e.g. spreadsheets, symbolic mathematics programs, data logging, C programming (C compiler and C environment)? Is the software the tool needed to complete the course? Does the software cover the whole syllabus (or chunk as appropriate)? Is the software compatible with available hardware?
- Training and Support - How familiar are the students with the software? How familiar are the staff members with the software? Is the software self starting for students (not

> waste staff-to-student contact time teaching students how to use software)? Are there on-line tutorials (same as above)? Is the supplier of software stable (important if the software is very expensive - security and worthwhile investment)? Is the main purpose of the software to develop transferable skills (i.e. required in industry and future life)? Is the software the best representative of the best practise in industry?

However it is still important that the software being used in courses is the most up-to-date on the market in order to maintain student interest.

Maintaining software

The length of time that software can be used on a course is so short for three reasons - technology advances so quickly, subject matter needs updating periodically and material can use features that are not supported on later versions of technology[2]. For these reasons the web has a major advantage as a learning tool when compared say to CD-ROMs as it is more flexible since it can be regularly updated. It is therefore vital that the means to devise plans and procedures for updating material is established when the course is set up.

Conclusion

There is such potential for enhancing courses by the introduction of suitable computer based education (CBE) e.g. software. For instance, computers have several advantages that deal with previously impracticable problems (do things faster and store large amounts of information for easy access), attempt to solve new problems that were previously considered impossible and used to consider questions that were not previously considered[2]. Using software provides the opportunity to introduce material to students from a different perspective than traditional methods of presentation[4]. It is very easy to want to include the latest material into a course without really considering how it benefits the course e.g. software, new textbooks, equipment and visits. It is important to decide whether the presence or absence of that material is the best thing for that particular course. The author of a course needs to provide a clear list of useful sites (e.g. search engines) and links (e.g. CAL materials) on the web for students to increase enthusiasm and account for a variety of thought paths[5]. It is important to realise the layout of pages on the web affects clearly how much information is understood and whether information can be accessed. It is better to keep things simple with links to the correct pages, than pages that look great but are difficult to understand. There is still a place for traditional lectures, but with the addition of software both in and out of the lectures (email, computer conferencing, face-to-face discussions, demonstrations of models of physical situations). Email can be used to help produce a list of the most commonly asked questions that can then be displayed on the web. Preparation and forward thinking is a pre-requisite to any lecturer using software as part of their course[3]. The rapidly falling cost of technology compared with conventional teaching methods[2] is another justification for making the use of software a more integral part of courses.

As the use of computers becomes more common future students will need less support and encouragement to use software as part of their courses. I feel that the present day students expect computers to be used as part of their courses (inevitable part of university life[8]) and consider it an extra life skill. I think that new students would usually be willing to learn extra computer skills outside of their course, on a short-term basis, to improve their skills for their future, as well as to enable them to use software as part of their course. I do not believe that undergraduates are as lacking in computer skills as was considered a few years ago - this is indicated by the number of homes with Internet access. There are a variety of courses that can be accessed via the web that can enable students to gain skills that they can utilise in their courses. In order for students to be involved in deep learning it is important to promote more active student centred learning. Remember that you are trying to help your students gain knowledge and increase their understanding - so don't just spoon feed them - help them to become independent.

References

1. Reid, T. A. (1993) The Benefits of Using Plato and CBE Generally. Internal paper, The University of Western Australia, October.
2. Reid, T. A. (1994) Perspectives on computers in education: the promise, the pain, the prospect. *Active Learning* **1**, 4-10.
3. Proctor, A. and Donoghue, D. (1994) Computer based assessment: a case study in geography. *Active Learning* **1**, 29-34.
4. Peel, A. (1994) Computer based assessment through hypermedia. *Active Learning* **1**, 35-37.
5. Parrington, N., Ferguson, I., Hedges, S. and Spence, L. (1994) The use of hypertext and the world wide web in teaching and assessing software engineering. *Active Learning* **1**, 39-42.
6. Whalley, W. B. (1995) Teaching and learning on the Internet. *Active Learning* **2**, 25-29.
7. Hicks, J. (1995) Some active learning implications of the information era. *Active Learning* **2**, 39-41.
8. Duffy, C., Arnold, S. and Henderson, F. (1995) NetSem - electrifying undergraduate seminars. *Active Learning* **2**, 42- 48.
9. Davies, M. L. and Crowther, D. E. A. (1995) The benefits of using multimedia in higher education: myths and realities. *Active Learning* **3**, 3-6.
10. Bacon, R. (1996) The effective use of computers in the teaching of physics. *Active Learning* **4**.
11. Drury, A., Gladwin, R. and Walker, S. (1996). Implementing CAL: success stories from chemistry. *Active Learning* **5**.
12. Tait, B. (1997) Constructive Internet based learning. *Active Learning* **7**.
13. Oliver, M. and Conole, G. (1998) Evaluating communication and information technologies: a toolkit for practitioners. *Active Learning* **8**.
14. Joyes, G., Choo, B. S., Pavic, A., Newman J., Beeby, A. and Wainwright, P. (1998) Learning outcomes and the role of evaluation in Embedding COMPACT. *Active Learning* **8**.
15. Nicol, J. (1998) Tools for computer assisted assessment. *Active Learning* **8**.
16. Lee, J. and McKendree, J. (1999) Learning vicariously in a distributed environment. *Active Learning* **10**.

[Active Learning is available on-line at http://www.cti.ac.uk/publ/actlea/]

Simone Richardson
CTI Centre for Physics
University of Surrey
Guildford
Surrey GU2 5XH
United Kingdom
S.Richardson@surrey.ac.uk