Co-operative cross-platform

courseware development

R. D. Harding, S. W. Lay and H. Moule

Department of Applied Mathematics and Theoretical Physics,
University of Cambridge

The UKMCC (UK Mathematics Courseware Consortium) is a Consortium funded under TLTP (Training and Learning Technology Programme) to produce courseware for service mathematics teaching, using the SEFI (Société Européenne pour la Formation des Ingénieurs) syllabus. There are agreed courseware design guidelines and a simple courseware management system which allows cross-referencing. Courseware is divided into modules, with an author as implementer for each. On any one hardware platform, a variety of authoring languages is possible. Across hardware platforms, the design guidelines ensure that conversion is possible, and will preserve look and feel. We argue here that these arrangements provide a basis for continued co-operation between authors and future development as the technology changes..

Introduction

Of all the TLTP projects, the UKMCC (UK Mathematics Courseware Consortium) has involved the largest number of institutions in authoring courseware. This was part of a deliberate attempt to set up an 'open courseware' system which can continue to be adapted and grow even after the end of TLTP funding. In turn, such an objective requires a cross-platform approach to courseware development. Even when considering the academic community across a single subject at just one time, there will be a variety of hardware and software that people will want to use, but a ghost of a case can be made out for attempting some sort of standardization. When change over time is considered, it is inconceivable that hardware and software platforms will remain unchanging, and one way to manage changing standards is to have a courseware structure that permits variety. This allows different parts of a collection of materials to evolve at different speeds.

The UKMCC is writing mathematics courseware, but all the issues raised above will arise in many other disciplines. This paper will also deal briefly with three other issues, which although general in themselves may require different solutions in different disciplines:

• Who should write courseware, academics or programmers?

- How should courseware be used?

- What is the future of courseware?

The UKMCC

Even before the TLTP was announced, the learning-technology community in university mathematics was considering ways to secure wide involvement in courseware writing. This was a result of a large collaboration under the older CTI (Computers in Teaching Initiative) (Harding, 1993), in turn leading to the setting up of the CTI Centre for Maths and Stats (CTICMS) at Birmingham. Two pilot projects (Maths 2 CD-ROM, 1989–1991; Harding, 1994) had gathered sufficient experience and demonstration materials to attract recruits for the UKMCC. Over 30 authors (see Acknowledgements) from 26 institutions joined together to form the UKMCC, and an early account of the UKMCC is given in Beilby (1993).

Phase 1 of the UKMCC effectively spanned two and a half years from Autumn 1992 to Spring 1995. During this period, a basic authoring style was established and 30 modules were commissioned from authors, to cover the basic topics in the SEFI (Société Européenne pour la Formation des Ingénieurs) syllabus. The SEFI syllabus is primarily meant for engineering mathematics, but it is a comprehensive international standard and has met with general approval amongst members of the UKMCC as a suitable syllabus for service course mathematics in the sciences. Phase 2 is nominally from April 1995 to the end of March 1996, during which time a further 20 modules are planned, although because of the way that authors plan their time, these will not all be complete until September 1996. A module covers approximately the equivalent of four to five hours of conventionally lectured material.

Who writes courseware?

In planning a project to deliver such a large amount of material, there are two poles to the strategies that might be used. One is to use academics as advisers or specifiers of materials and have the implementation done by technical staff. The other pole is to have the academics as implementers. The UKMCC opted for a policy that is closer to the latter pole (authors as implementers) although resource centres specializing in each of the main platforms have been set up to provide technical support to authors.
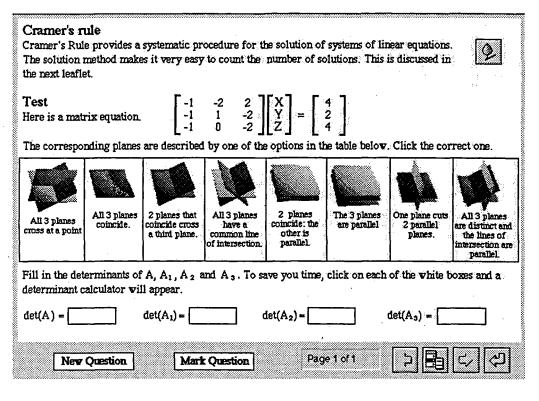
The UKMCC's main reasons for leaning towards the authors-as-implementers policy were partly to harness the enthusiasm that already existed amongst some academics for courseware authoring, and partly to help increase the level of experience in the academic community in a way that would be retained for the academic world at the end of the project. An additional reason is an assumption that academics are more likely to use courseware when they have contributed to it themselves, and know that they could change or adapt it themselves as and when they want.

Mathwise and the Courseware Manager

One consequence of the authors-as-implementers policy is that any project adopting it must either be prepared to take a draconian line on which authoring platform to use (in regard to both hardware and software), or be prepared to accommodate the usual range of prejudices and

preferences amongst academics in computing matters. To involve as much of the subject community as possible, the UKMCC wanted to accommodate almost any preference, but there were two provisos. The first was that there had not to be so much diversity that student users were confused (by different interface conventions for example), nor had this to hinder courseware written by one author being referenced from materials written by others. The second proviso was that in a project with finite resources it was not feasible to offer support for an unlimited variety of platforms.
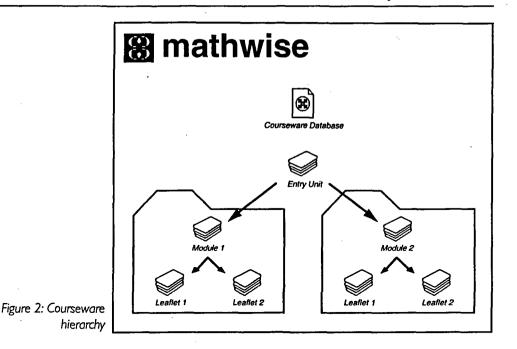
The consensus that was reached within the UKMCC was to use PC and Macintosh hardware, and Toolbook, HyperCard and Authorware as software for authoring. Software was thought to be of greater long-term significance than hardware choice, as industry is developing hardware which will run any major operating system suitable for personal systems, and we are also seeing more packages (like Authorware) which work across platforms.

To contain the natural diversity of authoring styles to an acceptable level, and to create a look and feel to be the hallmark of UKMCC modules, a document known as the *Courseware Design Specification* was produced. This was based on the experience of the Renaissance (Harding *et al*, 1995) and Nuffield (Harding, 1994) projects, developed in discussions with all UKMCC partners. Plans were drawn up for a supporting environment within which modules would run. The whole system of environment plus modules was given the name Mathwise. Figure 1 shows a typical screen from a Mathwise module: *Solutions of Linear Equations*.

Figure 1: Cramer's Rule

mathwise

Courseware Database

Entry Unit

Module 1

Leaflet 1   Leaflet 2

Module 2

Leaflet 1   Leaflet 2

*Figure 2: Courseware
hierarchy*

The UKMCC's original plans were to restrict authors to Toolbook, HyperCard or Authorware, and use the built-in facilities of these systems to cross-link between documents of different type. However, it became clear at an early stage that none of these built-in facilities would allow cross-linking to be done in a fully satisfactory manner. This led us to design a separate piece of software which can link together documents written under any application (on the Macintosh for example, such documents could be HyperCard stacks, Authorware packages, Microsoft Word files, Maple or Excel documents, etc.). This software we call the Courseware Manager; distinct versions are needed for PC and the Macintosh, but it has essentially the same functions on each. A fuller description has been published elsewhere (Harding *et al,* 1995).

To be usable under the Courseware Manager, a piece of courseware must conform to the simple structural rules contained in the *Courseware Design Specification,* as illustrated in Figure 2. There is now a TLTP-supported working group, EPOC (Enabling the Provision of Open Courseware), with one of us (Lay) as a member (URL: http://www.amtp.cam.ac. uk/icrd/EPOC/). Once a generally agreed open standard exists, Mathwise will want to conform.

Such a standard (Mathwise's own, then a fully open standard in the future) underpins the achievement of many of the UKMCC's objectives. Although for the present, UKMCC authors may prefer to stick with our major choices of authoring packages because help and advice is available for them, there is no technical reason why materials authored using other systems should not be included. In Phase 2, one author will be using SuperCard, and in the future authors can migrate to newer and better systems as they become available. Older courseware will remain compatible (although of course there is no magic way that it can improve itself!). User sites are able to customize Mathwise by removing, adding or replacing any courseware component that is visible to the operating system as a single file or collection of files. This

process may, of course, require some authoring work by the user site, and the use of the Courseware Manager's administrative facilities.

Learning style

Mathwise can be used in any of the following ways: in place of lectures, as partial replacement for lectures, as an addition to lectures, or as a library resource.

Institutions will need to decide how to manage the use of these materials, and incorporate advice and directions for use into their standard procedures. Students' access to the materials also needs consideration. There could be central or distributed labs, with scheduled hours, bookable individual seats, or just free access. There could be network access via students' own machines in their halls of residence.

Our courseware design and the Courseware Manager itself are intended to make it easy for lecturers to configure Mathwise modules, and any other courseware materials required that conform to the courseware standard.

The future of courseware

The basic character of any body of courseware, its ease of use, modularity, availability across campus, adaptability to local conditions, etc., is inextricably linked to the style in which courseware will be used. Unfortunately, it seems to us that the TLTP has not addressed the issues adequately: not only should the demand for courseware be addressed (as discussed further below) but, in addition, career issues such as incentives and support for staff to introduce new techniques into their teaching need to be considered.

Improved effectiveness of learning has been a major objective of educational computing since CAL first became a technical possibility. However, there is very little evidence from experience or in the literature that computers can replace human teachers in higher education, and the evidence points to computing costs being add-on, and the benefits being enhancement of education rather than replacement of older methods. Hooper's final report (1977) on the NDPCAL nearly 20 years ago made this clear, and although since then the technology has made huge advances, the basic modes of use of computers in teaching and their likely effectiveness has remained much as summarized by Hooper. It is disheartening that either the lessons of the NDPCAL were not known to TLTP policy makers, or that no public debate was conducted to explain why it was thought that dramatic changes were imminent.

Supporting the authoring of courseware is only half of the equation, the supply side. Consortia like the UKMCC have no control over the adoption and use of their products. It is necessary to make an informed guess at all the ways institutions will use courseware and try to write appropriate materials, but no project can guarantee that materials will be adopted. There needs to be a demand-side policy as well, and it is here that the signals from HEFCE are most confusing. Assurances have been given that successful efforts in courseware development will be recognized in research and teaching assessment exercises, but there is scant evidence that lecturers feel there has been any change in the link between their efforts in this direction and career development. Perhaps ALT has a vital role to play here?

Acknowledgements

References

Beilby, M. (1993), 'Mathwise: a learning environment (part 1)', *Maths & Stats*, 2 (5) (November).

Harding, R.D. (1993), 'Computer interactive texts' in *Proceedings of 4th Annual Conference on Technology in Collegiate Mathematics* (Reading, MA, Addison-Wesley), 132–8.

Harding, R.D. (1994), 'Nuffield interactive mathematics courseware', *Computers and Education*, 22 (1/2), 73–82.

Harding. R.D., Lay S.W., Moule, H. and Quinney, D.A. (1995), 'Multimedia interactive mathematics courseware', *Computers and Education*, 24 (1), 1–23.

Harding, R.D., Lay, S.W. and Quinney, D.A. (1995), 'A consortium approach to courseware design in mathematics', *Computers and Education* (Proceedings of CAL 95) (in press).

Hooper, R. (1977), *National Development Programme in Computer Assisted Learning. Final Report of the Director*, London, Council for Education Technology.

Maths 2 CD-ROM (1989–1991), published by UltraLab, Anglia Polytechnic University, produced under the Renaissance Project, sponsored by Apple Computer UK.