Descriptive account

Challenging the Mortality of Computer Assisted Learning Materials in the Life Sciences: The RECAL Project

Rachel Ellaway, David Dewhurst and Stewart Cromar

Learning Technology Section, College of Medicine and Veterinary Medicine, The University of Edinburgh, EH8 9XD

Date received:31/03/2004 *Date accepted:23/04/2004*

Abstract

The development and use of computer assisted learning (CAL) materials in the life sciences is well established and, in the UK at least, significant resources have been provided to enable this. Some years on from when the major investments took place teachers are facing the problem that the technologies used to develop and deliver the CAL programs have become obsolete in the face of rapid and constant changes in desktop computing. The content and pedagogical design of these programs were intrinsically tied to their delivery technologies such that when the technology becomes obsolescent the programs are either abandoned or redeveloped again and again at a significant cost in time and resources.

The RECAL project, based at the University of Edinburgh, is developing methodologies that make use of new ways of abstracting and managing a CAL program's assets, pedagogical design and run-time components to allow for much greater longevity and flexibility of such materials.

This paper is an introduction to the RECAL approach, its use of common standards and specifications for describing materials and educational activities and how this can benefit developers and users of CAL materials.

Keywords: CAL, XML, reuse, standards, change

Introduction

> *"computer hardware and associated software depreciate quickly … partly because computers are somewhat fragile but mostly because of the astonishing rate at which technology has developed" (Landauer, 1995, p115)*

Over the last decade the focus of the use of technology to support teaching and learning in the life sciences has been to produce interactive multimedia computer-assisted learning (CAL) programs to support specific areas of the curriculum. There have been a number of strategic government-funded initiatives in the UK to stimulate the use of technology in higher education notably the Computers in Teaching Initiative (CTI), the Teaching and Learning Technology Programme (TLTP), and the Learning and Teaching Support Network (LTSN). TLTP in particular made over £30 million available in two phases to fund largely discipline-based projects which resulted in the development of some excellent courseware in areas such as pharmacology, veterinary medicine, and physiology. A typical example would be 'pharma-CAL-ogy', a TLTP phase 2 project (1994-97) involving a consortium of UK university pharmacology departments led by the University of Leeds, which developed a range of interactive CAL tutorials, CAL simulations of laboratory experiments and videos to support pharmacology teaching. Many pharmacology departments, particularly in the UK, have acquired at least some of these materials and made them available, via university-wide networks, as support resources. However, there seems to be little evidence of them being integrated into courses (Markham et al, 1998) despite the results from a number of studies, in which the usefulness of such programs has been evaluated, and which suggest that using such resources to replace traditional modes of course delivery can be effective (Dewhurst, et al, 1994; Hughes, 2001; Fawver, et al, 1995; Guy & Frisby, 1992; Leathard & Dewhurst, 1995; Johnson, Dewhurst & Williams, 1997; Dewhurst & Williams, 1998; Dewhurst, Macleod & Norris, 2000, Hughes, 2001). There are a number of reasons for this: most CAL programs are not web-deliverable and only available over a local area network (LAN); the content is locked within the authoring shell and therefore not editable to suit local needs; teachers don't have the time and/or skills to integrate the CAL programs into the curriculum (essential if they are to be useful (Hughes, 2000)) and there is no institutional recognition for their effort; there is resistance to using resources developed by others; most CAL programs are intrinsically linked to the technologies used to develop and deliver them, technologies which are rapidly becoming obsolete rendering the resources useless.

The usefulness of a CAL program depends on the closeness of fit of the program with the needs of the teacher and their willingness to adopt materials developed elsewhere. Although the needs of different higher education institutions are similar, each values the differences that identify their particular courses as being different and, in their view, superior to others. This academic diversity and institutional conservatism is a characteristic of universities worldwide and is often a barrier to introducing teaching materials that are 'not invented here'. Thus, a CAL program which addresses the learning goals of, for example, a specific animal lab may be the perfect solution for one university (usually the one which developed it) but less than perfect for others. Under these circumstances it is clear that although a proportion of the content of the CAL program will be suitable for many universities, it will be rejected as it fails to meet their needs.

Anecdotal feedback suggests that teachers would like to be able to use those components of the alternative CAL program which they like but they would also like the facility to add some of their own i.e. customise to their own needs. Thus for instance, in a pharmacology simulation program they might wish to add new or delete current test drugs, extend the dose range, add new tasks or delete current ones, display the data in a different format (e.g. extend the timebase), translate the text into a language other than English, add formative assignments etc. Similarly small clips from videos of animal dissection or anatomy might be more useful and usable than the whole. To date, the constraints of the authoring tools used to create the alternatives make this very difficult and expensive: institutions would have to purchase the software, and they would need staff with the skills, time and experience to develop a learning resource, and this would be repeated in every institution that needed a variation on the basic CAL.

Usually a multimedia CAL program would be developed by a small team comprising a content expert (e.g. an academic pharmacologist), a learning technologist and an educational designer adept at transforming traditional materials into an interactive format suitable for on-line delivery and student-centred learning. The content provider and the educational designer would assemble the components of that material – write the text, define what media they needed, find data traces, create self-assessment activities, suggest ideas for animations, produce video/audio clips etc. to produce a storyboard. This would then be developed into a stand-alone CAL package by the learning technologist using a multimedia authoring tool such as Assymetrix Toolbook or Macromedia Authorware or, before such tools were available, programming shells such as Basic or Hypercard. This process is time-consuming and therefore expensive, and results in a CAL program in which all of the components are 'locked into' the application and linked intrinsically to its delivery mechanism. While the educational content and learning design may still be valid, often changes in technology (such as the move from DOS to Windows, from 16 bit to 32 bit processing and from VGA to XGA screen resolutions (640x480 pixels to 1024x768 pixels)) result in the delivery mechanism becoming obsolete, the consequence of which is that the content and learning design are lost too.

Until now, the only options have been to either abandon these obsolete materials, or redevelop them again and again at significant cost in both time and staff resources. Institutions, in their efforts to provide staff and students with an up-to-date technology infrastructure, often take little account of the effect upgrading the technology will have on the use of educational software and teachers are forced to rely on outdated, often unsupported technologies, to run the legacy software. Persuading teachers to use technology to support teaching and learning is difficult enough without this added barrier.

The problem, expressed simply, is that the pedagogical design, media assets and run-time delivery mechanism are intrinsically locked together so that the failure or obsolescence of one component dooms the whole. Interestingly, the technological changes that have caused this core obsolescence problem have also now provided ways to circumvent such problems in the future. The RECAL project described here is developing methodologies that make use of new ways of abstracting and managing the educational content (assets), pedagogical design and run-time components of existing bioscience CAL materials to allow for much greater longevity and flexibility of such materials. This paper is an introduction to the RECAL approach, its use of common standards and specifications for describing assets and educational activities and how it can 'future-proof' CAL materials and thus benefit developers and users of CAL materials.

The RECAL Project

Harnessing new technologies – Flash and XML

It is sometimes worth remembering how recent a phenomenon the mass use of the world-wide web is. With the growth of the web most development platforms became able to deliver web-enabled versions of applications developed within them. The bandwidth constraints intrinsic to the web required a move to much lighter and faster ways of handling multimedia content; vector rather than pixel graphics, streaming audio and video and the modularization of applications so that only the required part of the CAL was downloaded. In response, new technologies were developed that were principally focused on addressing these issues, Macromedia's Flash being particularly dominant among them. However, despite much innovation in developing fast and light multimedia technologies, probably the most revolutionary innovation was the introduction of a new format for text files; 'extensible markup language' or XML.

XML differs from hypertext markup language (or HTML - the bedrock of the web) in that the markup is not predefined (markup is the tags in the file that describe the content, for instance in HTML <b> indicates bold and <p> indicates the start of a paragraph). In XML user-defined schemas of markup tags can be used to describe and structure material in very powerful and specific ways. XML is a text format and as such is completely cross-platform and able to be created and transported rapidly between applications. The ability to read XML is an increasingly common feature of all web technologies and it is the use of XML in preserving and reusing the content and design of CAL materials that can expand their lifespan by several orders of magnitude or more. However, the ability to define the markup in XML has the potential to make it utterly specific to a local application. The power of XML is enhanced when a user community adopts and shares a common specification for how their XML will be formatted. This has been addressed by the burgeoning work on learning technology standards and specifications.

The issues of obsolescence and redundancy are not new to learning technology. These, along with those associated with interoperability problems between different applications or operating systems, have been driving an international agenda for a number years now to establish common standards and specifications for interoperability between applications and systems describing learners, assessment, media and latterly pedagogy (see further reading). These standards and specifications are largely based on XML as a common intermediary language with each standard or specification providing a set of rules for assembling and formatting the XML in describing the entity or process that they cover.

There are a number of key specifications that are appropriate to CAL materials. However there are, in some circumstances, several specifications that cover the same areas, in particular the ADL/SCORM model, IMS Simple Sequencing and IMS Learning Design. Which one is most appropriate depends on where you are and what you want to do. For instance ADL/SCORM, because of its links to the US government, is dominant in North America, while the IMS specifications have more currency in Europe. SCORM and Simple Sequencing only model individual learners while Learning Design is the only one of these three that can model collaborative learning activities.

The RECAL Architecture

RECAL is a fusion of new fast and light web-based multimedia technologies and the use of XML based on common learning technology standards and specifications. The architecture is based on a number of disassociated components, each of which can be modified independently, being assembled dynamically when the CAL is run. This is illustrated in figure 1.

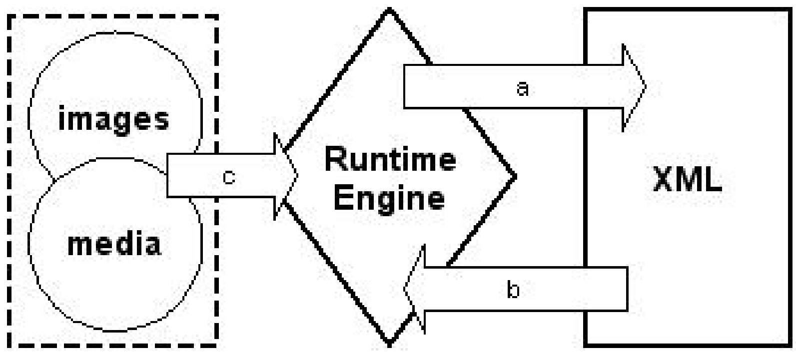

Figure1: the RECAL architecture. When the CAL is started the runtime engine looks for the XML parameter file (a) and, finding it, loads the instructions and content specified in the XML (b). Part of these specifications are URL pointers to external resources such as images and sound files which are loaded in to the runtime as well (c). [fig1.jpg]

Any of the components should be able to be changed independently of the others – images could be added or edited, CAL sections extended or a different runtime engine used. Thus a CAL program can be tailored to the local circumstances in which it will be used. This is shown in figure 2 where a different XML file and different resources are pulled in to the same runtime engine.

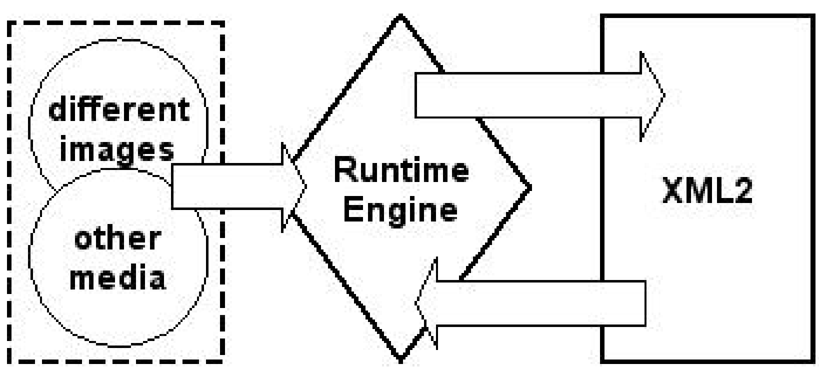

Figure 2: changes to one or more of the RECAL components allow the CAL program to be adapted to a specific context. The new instructions and/or media are only loaded at the point where the CAL program is executed. [fig2.jpg]

The RECAL Process

The RECAL project has been set up to explore the ways these new approaches to developing and maintaining CAL resources can improve their lifespans and at the same time make the authoring, development and management phases much simpler and transparent. This project will focus on:

1. Disaggregation of existing CAL programs designed to support teaching and learning in pharmacology and developed by the authors. This will unlock the content and release the media assets such as images, diagrams and animations, information assets such as text, data and algorithms, and activity designs such as the structure, organization and rules the CAL programs contain. In some cases the CAL programs are still available in an uncompiled form, allowing direct access to these components, in other cases the components have be extracted by taking screengrabs, transcribing text and by recreating those components where no direct extraction is possible. Animated sequences usually need to be recreated even if access to the source is still available as the means for animation is almost always tied to the particular development platform used. There is also a round of quality assurance involved at this stage. For those CAL programs where materials do not meet contemporary standards of quality, these materials need to be recreated. This will most likely be the case for visual assets such as diagrams, photographs and animations.
2. Object management - cataloguing and management of these disassembled resources. Each component needs to have metadata associated with it indicating its basic properties and its provenance. This includes aspects such as the object's file type, location, keywords, intellectual property rights, provenance, location, and size. The resulting objects and metadata are then to be managed in a learning object repository, designed to specifically hold and manage learning object metadata. This stage will also involve the development of a simple workflow, supported by a set of web-based tools, to enable teachers to catalogue and add appropriate descriptive metadata to the objects so that they can more easily be found in the repository, thus aiding resource discovery.
3. Pedagogical design - rendering the pedagogical design of the original CAL programs into either IMS Simple Sequencing or IMS Learning Design XML. IMS Simple Sequencing, in that it consists of ways to describe a learner's activity and how their progress is controlled based on this activity, is most like traditional CAL programs with their sequencing rules covering aspects such as branching or conditional advancement. Learning Design on the other hand is more abstract but far more pedagogically focused. At this stage RECAL is using IMS Simple Sequencing.
4. Run-time engines - the development of run-time tools that can read the XML, call down the appropriate resources from the repository and provide the interface for the user. RECAL is principally using Macromedia Flash for the standalone run-time shell but is also trialling web rendering in plain text to support users with visual disabilities. The run-time engine needs to be able to load and parse the XML instructions, load the appropriate templates and populate them with the appropriate media assets. It is expected that over time new run-time applications can be built for new platforms or applications. The term 'run-time engine' is being used to describe an application that can adapt and change depending on its imported instruction set.
5. Authoring tools are being developed that can create or adapt appropriate CAL-oriented XML for the run-time engines to use. The goal is to provide

academics who use these packages with an editor that lets them customize the XML instructions and therefore to customize the CAL program for their local needs. Customization may include the reorganization, addition or removal of sections, images or text. It may involve changing the units of measure, such as drug dose, or the names of entities, such as drugs or electrical stimulation parameters, or it may involve adding or changing questions and tests. More radically still it could involve a complete translation of the program's text into another language or reconfiguring the program to meet the needs of a different educational context. The authoring tool will create well-formatted XML text files for the run-time engine to use and will use a simple graphical interface to support the majority of authors who will be unfamiliar with XML. As an exemplar of this kind of approach see the RELOAD project at www.reload.ac.uk.

6. Expanding the community of developers. Having established the infrastructure described above, the aim is to expand the RECAL franchise of academic authors to include a number of developers in Europe, working initially on translating the XML instructions to different languages. They will have full access to the object repository and by working collaboratively it is anticipated that this network of authors will exchange and develop transparent, portable and long lived CAL materials.

RECAL Pilot

A proof-of-concept pilot of each of these stages was completed in the autumn of 2003. A Flash run-time was constructed that could dynamically load and render an external XML parameter file and the resources it identified. The pilot has only involved one CAL program, an English-language version of a program originally developed for DOS, later rewritten for early versions of Windows using Assymetrix Toolbook version 2.0 and redeveloped again in 1998 for later versions of Windows using Macromedia Director.

The program was disaggregated and the XML was assembled in a text editor. The XML format developed for the pilot was not at this stage specifications-compliant and was developed by iterative cycles of matching functionality to coding. Screen grabs from the pilot can be seen in figures 3a and 3b.

A sample of the pilot XML can be seen in Table 1.

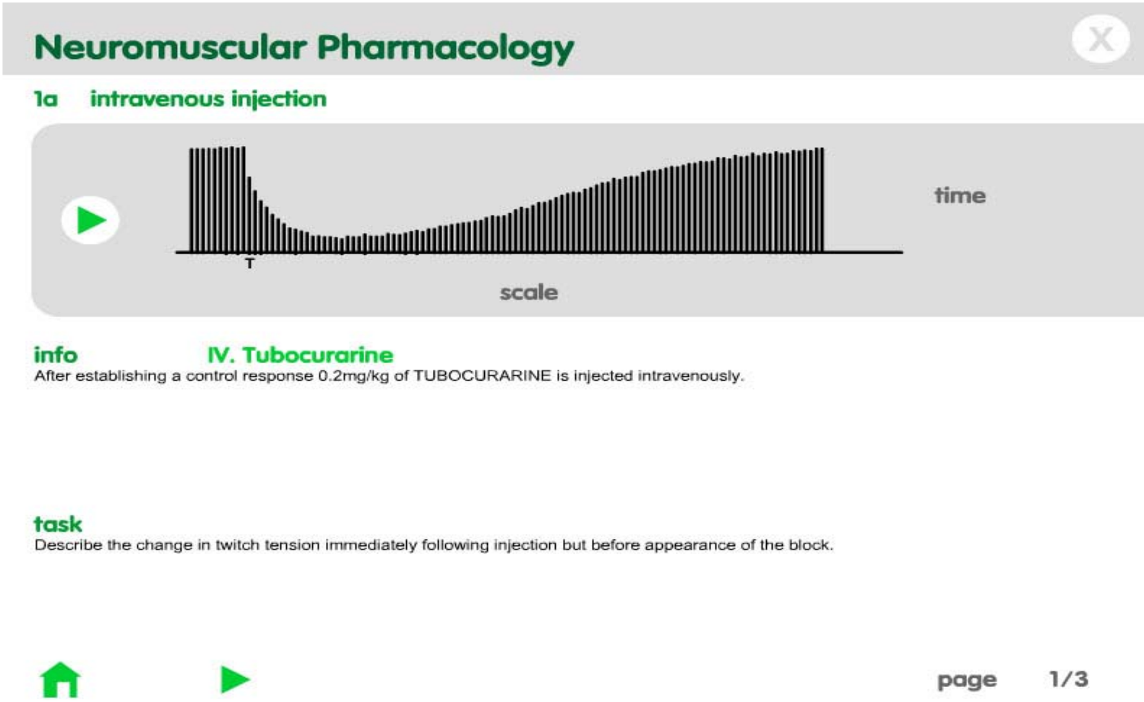

Fig 3a: This screen grab illustrates a typical 'experiment' page type with multiple flash and text components.

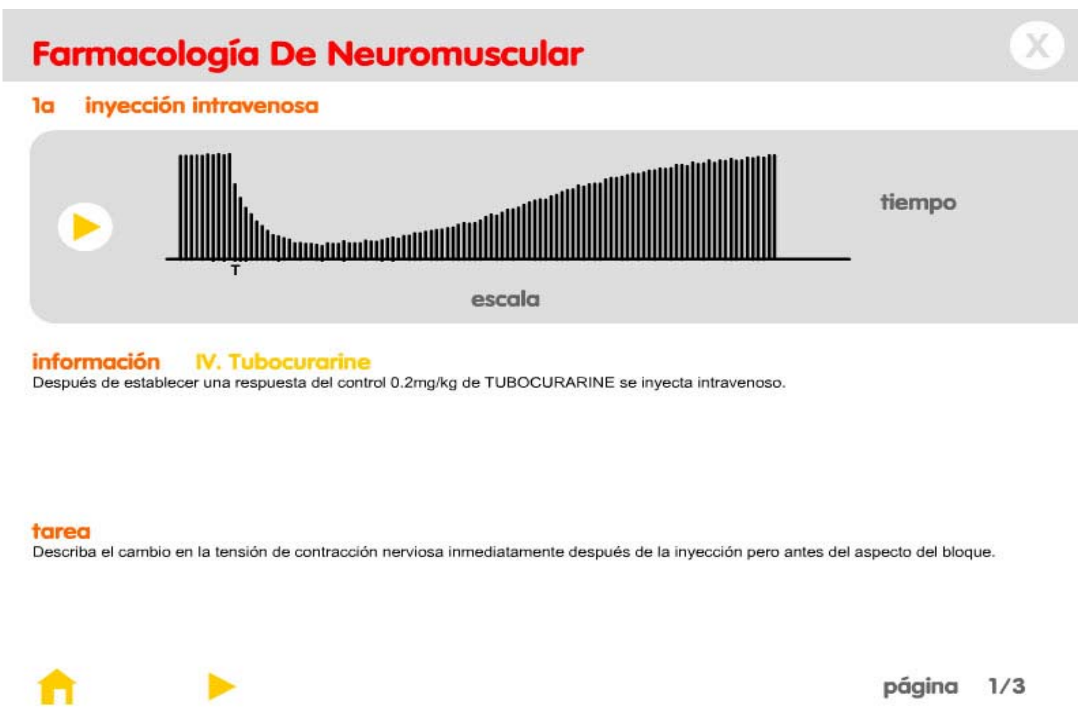

Fig 3b: The same experiment page, but using a Spanish translation of the original text file.

Table 1: The left column shows the complete XML for the 'experiment' page in English shown in Figure 3a. The right column indicates which elements of the XML code required Spanish translation to produce the page shown in Figure 3b.

ENGLISH	SPANISH
<page ptype="experiment">	<page ptype="experiment">
<id phase="1">1a</id>	<id phase="1">1a</id>
<subtitle>intravenous injection</subtitle>	<subtitle>inyección intravenosa</subtitle>
<image type="flash" src="images/1a.swf" />	<image type="flash" src="images/1a.swf" />
<info drug="IV. Tubocurarine" >After establishing a control response 0.2mg/kg of TUBOCURARINE is injected intravenously.</info>	<info drug="IV. Tubocurarine" >Después de establecer una respuesta del control 0.2mg/kg de TUBOCURARINE se inyecta intravenoso.</info>
<task>Describe the change in twitch tension immediately following injection but before appearance of the block.</task>	<task>Describa el cambio en la tensión de contracción nerviosa inmediatamente después de la inyección pero antes del aspecto del bloque.</task>
</page>	</page>

During the pilot's short development cycle several screen design and Flash specific technical issues became apparent:

- rather than modify the original program content to fit within one generic template, multiple page types were developed to suit the requirements of each interaction. It is expected that a number of sub-templates of this kind will be developed to provide a suitable user interface for each of the interaction types. For instance, the experiment shown in Figures 3a and 3b would constitute one sub-template, while the associated MCQs would use another.
- in an effort to simplify end-user editing, text-based navigation was kept to a minimum through the use of common web-browser iconography. For example the Home button and the back/forward buttons resemble closely the navigation used in Microsoft's Internet Explorer.
- in addition to explicit language issues, e.g. translation between English and Spanish, there was also an issue of the encoding language for the XML. Whilst in development both a text editor and a dedicated XML editor were used. It only became apparent after a number of problems had been encountered, that the two programs were using different XML encoding (see further reading).

A short round of in-house testing and evaluation was carried out with the purpose of tailoring the pilot application towards non-technical academics. Simple authoring tasks included editing text, re-ordering the existing question bank, and adding new questions. The outcome of this testing period was a set of tools accessible to both authors and developers and a clear path to a scaled up and specifications-compliant next stage.

RECAL Elements

Any given CAL program is likely to have different interactive elements within it. One program may have a series of text, animation and multiple choice question (MCQ) elements, while another may consist primarily of an algorithmically-driven experimental simulation. In disciplines such as pharmacology, physiology and biochemistry there is a strong element of practical laboratory teaching some of which may be supported or replaced by CAL simulations, many of which are based on mathematical models which are updated in real or simulated time.. Similarly, an animation sequence which depends on user input could be included as a small part of a CAL program; for example, a titration which allows acid or alkali to be added and calculates the result mathematically depending on which of these the student chooses and the amount or rate at which it is added. The flexibility in terms of supporting user selection of these variables is central to the way these programs are used to teach and the use of RECAL in these disciplines would be severely constrained if it were not able to handle these more complex elements.

RECAL works by identifying each of the forms of presentation and interaction as RECAL Elements. . Each Element can be instantiated and controlled from external XML parameter files within a RECAL application. Thus, a RECAL application will have its constituent Elements defined and organised in one parameter file and each RECAL Element within it instantiated and running from subordinate parameter files. Currently, RECAL has been used to develop learning activities from existing CAL programs in which there are no data traces generated from algorithms. However, it should be possible for RECAL to handle more complex interactions such as algorithmic simulations but only if the original algorithms and variable parameters are still available. In most cases when a CAL program has been published it has been as a compiled version for runtime delivery from which the underlying algorithms are no longer identifiable. RECAL can only recreate dynamic elements such as these if the underlying algorithms are known. Reverse engineering might be possible with the assistance of a good subject specialist but in that situation it might be just as easy to develop a new version of that element. So far the focus of the project has been on repurposing elements from existing programs but we would also like to encourage individuals or groups of teachers to participate in the project and develop new RECAL Elements of their own to extend the RECAL application further. Thus, the addition of a new drug to a pharmacology program would be possible whether it involved adding a new, defined data trace or a new predictive algorithm and a set of parameters which would dynamically produce a data trace in response to user-selected variables.

Supporting the user adaptation of algorithmic RECAL Elements, in particular, will need to be done with care and may be prohibitively complex unless the options for adaptation are limited. RECAL has not yet undertaken Element building of a full algorithmic simulation but does not underestimate the potentially Byzantine nature of such developments. Adaptation may be limited to ranges and forms of inputs rather than new drugs or other major parameter changes.

Discussion

The methodology adopted by the RECAL project has the potential to provide the academic authors, developers and managers of CAL resources with a much better return on investment (ROI) than has previously been possible with such materials. It also has the potential to allow the resources and educational designs that go to make these CAL programs to be more easily reused and repurposed and it has the potential to make them more adaptable and customizable to specific and varying educational settings. The opportunity that this type of approach offers for an institution to better manage teaching and learning resources centrally should also not be overlooked. For all of these reasons the authors wish to share and exchange their ideas and practices with the community to establish whether the potential can be realized, and if so, in what ways.

It is unlikely however that this will be straight forward as there are a number of issues that need to be addressed before any large-scale implementation can be established:

a) *Transparency and Complexity*
A CAL program's pedagogical design, when rendered in XML, can be complex and opaque to the untrained eye. The RECAL methodology requires intermediary tools for authoring and rendering the XML to allow non-specialists to work with CAL materials. Furthermore, there are major issues, discussed earlier, regarding the interpretation of specifications, the variability of pedagogical designs and the interpretations made by the run-time engine. Within a single project such as RECAL these can be accommodated, but in the community as a whole these issues have the potential to be as divisive as they are inclusive.

It is a common observation among software developers that complexity does not go away, it just changes from one form or location to another. It is important that the learning resources developed by teachers and the process of their creation are intellectually stimulating and provide at least as good an educational experience for students using this system as current CAL resources. Much of the development work rests on designing sophisticated run-time engines and, in order that the development of such engines does not become prohibitively complex, a degree of normalization is required, which in turn must inevitably circumscribe the variability and degree of adaptability possible from the RECAL approach in general. A pragmatic balance has therefore to be struck between the adaptability of RECAL and the ability to deliver a relatively simple and easy to use system.

b) *Stability and Change*
One of the main themes of the RECAL project is the use of specifications and standards-compliant descriptors for educational activities and resources. However, even specifications standards, change over time. Thus, although a specification-compliant expression of, for example a pedagogical design, might be hoped to be a solid and objective statement, in fact it is also subject to change. Changes to the XML information model can be accommodated more easily in the base XML than in an application that is using the XML. Thus the runtime engine is likely to be more volatile than the other components of a RECAL application.
It is important to remember that technological developments continue to outwit the pundits who predict what paths they will take. The authors therefore acknowledge that the claims for future-proofed and 'immortal' CAL resources need to be considered in the light of this poor predictability. However, XML does appear to be the best option currently available and the one that a very wide

range of systems and technologies are adopting for just the kinds of underlying reasons already addressed in this paper.

c) *Ownership and IPR*
The RECAL project will use existing CAL programs developed by the authors as a test-bed for the system and therefore IPR is not a major issue as all media assets and pedagogical design will be made freely available to academic colleagues to use in their own teaching. However, it still raises issues such as: if an academic changes an existing CAL program (which is the RECAL project's intent) then how much does the modified CAL program now contain aspects of that teachers IPR or indeed the institutions IPR? If over time a CAL program is adapted by a number of individuals what happens to ownership and IPR then? What if an academic uses media resources that they have (or have not) licensed from a commercial source?

The RECAL project raises significant although familiar issues regarding the reuse of learning materials. It has been recommended that a declaration of ownership giving scope, duration and conditions of use should accompany each resource (Duncan and Ekmekcioglu, 2003) when it is entered into the repository. This tracking of provenance is something that the RECAL project intends to support but there is every chance that once the CAL resources are 'out there' that authors will fail to provide such information.

d) *Reductionism*
Encoding human activity in a formal information model will involve decomposing it to its constituent parts. By doing so there is a danger that "all the factors that could be useful to the teacher are removed" (Koper, 2003). This is, however, less of a problem for CAL materials as their heuristics are most often based on an autonomous tutorial or practical session and a reductionist approach is therefore inherent in the authoring process. Whether this has knock-on effects in non-technological teaching practices as suggested by Hughes (Hughes, 2003) has yet to be seen, but it is likely that any significant improvement in the affordances of one method of teaching and learning is likely to have impact in other associated areas.

e) *Return on Investment*
There are a number of forms of investment involved with the use of CAL resources; technology, technology support, instructor and student costs (Collis and Moonen, 2001). The RECAL project will address the first two of these by making aspects of its architecture freely available, and it is hoped that the customization process can be made as intuitive as possible. The latter two are addressed more implicitly by empowering the academic to change CAL programs to meet their own needs using simple and intuitive tools, and by facilitating the closer integration and alignment of CAL programs to the course thereby making student activity more streamlined and focused.

f) *Why bother?*
Despite more than a decade of significant activity in developing CAL materials, the use of learning technologies remains patchy across the tertiary sector:

"there are still many academics who have not yet comfortably adopted technology in their teaching. So, if we have not yet solved the adoption issue, we should take care about forging ahead into the reuse domain" (McNaught, 2003)

The degree to which user modifications were possible to 'traditional' CAL materials has been minimal without major investment in staff (technical developers) and their associated tools. Even when this has taken place, it has tended to distance academics from the authoring and use of CAL programs. These issues, combined with the woefully short technical lifespans of CAL programs, has led to less than ideal returns from the time and expense invested in their development and use both by individual academics and by institutions as a whole.

However, if means were found to increase the useful lifespan of CAL programs and at the same time make them more adaptable to particular contexts of use and transparent to academic users, then it is to be hoped that the use of CAL programs can be made far more effective and efficient. RECAL is looking to address all of these issues by putting the tools directly into the hands of the academics, thereby empowering them and at the same time allowing them to change and adapt CALs to their own specific needs and contexts.

"the introduction of a technology 'specifies' the shape of a user's environment, but it is … a multilateral process where users are able to renegotiate the nature of that specification through their own user modifications" (Cornford and Pollock, 2003, p19)

It is hoped that a by-product of RECAL will be the opportunity for academics and students to retake control of at least some of their increasingly technologically-mediated environment and fashion it in accord with their needs and understanding of their learning and the subject at hand.

Further Reading

Hypertext markup language or HTML is a subset of SGML (standard general markup language) developed by Tim Berners-Lee in the late 80s and early 90s. It is now managed by the W3C – http://www.w3.org
Extensible markup language or XML is also a subset of SGML and is also managed by the W3C – http://www.w3.org

Although text files appear to be a common lowest common denominator format for exchanging information there are in fact many different encoding formats for text files. These usually have the same codes for the basic alphanumeric characters but differ for the various punctuation, glyphs and non-standard characters. For English-speaking users these are usually either 8 bit Unicode (UTF-8) or ISO Latin 1 (ISO-8859-1). There are similar but different encodings for other languages, such as Cyrillic (koi8-r).

IMS Global Learning Consortium, Inc. (IMS is the organization's name and is not an acronym) is developing and promoting open specifications for facilitating online distributed learning activities such as locating and using educational content, tracking learner progress, reporting learner performance, and exchanging student records between administrative systems. IMS is online at http://www.imsglobal.org/

ADL (the Advanced Distributed Learning) Initiative is sponsored by the US Office of the Secretary of Defense (OSD) as a collaborative effort between government, industry and academia to establish a new distributed learning environment that permits the interoperability of learning tools and course content on a global scale. SCORM (Sharable Content Object Reference Model) is the main ADL product and is defined as a Web-based learning "Content Aggregation Model" and "Run-Time Environment" for learning objects. – online at http://www.adlnet.org

CETIS (the Centre for Educational Technology Interoperability Standards) is a UK JISC-funded body acting as a broker and information resource in all areas of standards and specifications. CETIS is a member of IMS Global – online at http://www.cetis.ac.uk

Details on Flash and other Macromedia products can be found at http://www.macromedia.com

Communicating author Rachel Ellaway, e-Learning Manager, MVM Learning Technology Section, 15 George Square, Edinburgh EH8 9XD, UK: Telephone 0131 651749;Fax131651012;rachel.ellaway@ed.ac.uk

Bibliography

Collis, B. & J. Moonen (2001) *Flexible learning in a digital world: experiences and expectations.* London, UK: Kogan Page

Cornford, J. & N. Pollock (2003) *Putting the university online: information, technology and organizational change*. UK: Open University Press

Dewhurst, D.G., Hardcastle, J., Hardcastle, P.T. & Stuart, E. (1994) Comparison of a computer simulation program with a traditional laboratory practical class for teaching the principles of intestinal absorption. *Amer. J. Physiol., 267, (Advances in Physiology Education)**12** (1), S95-S103*

Dewhurst, D.G, Macleod, H.A, & Norris, T.A.M (2000) Independent Student Learning Aided by Computers: An Acceptable Alternative to Lectures? *Computers in Education: 223-241*

Dewhurst, D.G. & Williams, A.D. (1998) An investigation into the potential for a computer-based tutorial program covering the cardiovascular system to replace traditional lectures. *Computers & Education **31**: 301-317*

Dewhurst, DG, Macleod, HA, & Norris, TAM (2000) Independent Student Learning Aided by Computers: An Acceptable Alternative to Lectures? *Computers in Education **35**: 223-241*

Duncan, C. & C. Ekmekcioglu (2003) Digital libraries and repositories. In *Reusing online resources.* Ed Littlejohn A. H., pp135-145. London, UK: Kogan Page

Fawver A.L., Branch, C.E., Trentham, L., Robertson, B.T. & Beckett, S.D. (1995) A comparison of interactive videodisc instruction with live animal laboratories. *American Journal of Physiology: Advances in Physiology Education **259**, S11-S14*

Guy, J.F. & Frisby, A.J. (1992) Using interactive videodiscs to teach gross anatomy to undergraduates at The Ohio State University. *Academic Medicine,* ***67*** *(2) 132-133*

Hughes, I.E. (2000) Teaching Software; Will You Be Using A TLRP? *LTSN Bioscience Bulletin, Number 1, p4*

Hughes, I.E. (2001) Laboratory Practicals in Pharmacology Teaching - do they meet the learning needs? *Trends in Pharmacological Sciences,* ***22*** *(2), 71-74*

Hughes, I.E. (2003) Changes in the Use of Technological Methods of Teaching and Learning in Undergraduate Pharmacology in UK Higher Education. *Bioscience Education E-journal, Volume 1* *http://bio.ltsn.ac.uk/journal/vol1/beej-1-1.htm*

Johnson, M.I., Dewhurst, D.G. & Williams, A.D. (1997) Computer-based interactive tutorial versus traditional lecture for teaching introductory aspects of pain. *Association for Learning Technology Journal* ***5*** *(3), 22-31*

Kinzie, M.B., Strauss, R. & Foss, J. (1993) The effects of an interactive simulation on the performance and achievement of high school biology students. *Journal of Research in Science Teaching* ***30*** *(8), 989-1000*

Koper, R. (2003) Combining reusable learning resources and services with pedagogical purposeful units of learning. In *Reusing online resources*. ed Littlejohn, A. H., pp46-59. London, UK: Kogan Page

Landauer, T. K. (1995) *The trouble with computers: usefulness, usability, and productivity* USA: MIT Press

Leathard, H.L & Dewhurst, D.G. (1995) Comparison of the cost-effectiveness of a computer assisted learning program with a tutored demonstration to teach intestinal motility to medical students. *Association for Learning Technology Journal, 3 (1), 118-125*

Markham, T., Jones, S.J., Hughes, I. & Sutcliffe, M. (1998) Survey of methods of teaching and learning in undergraduate pharmacology within the UK higher education. *Trends in Pharmacological Sciences, 19, 257-262*

McNaught, C. (2003) Identifying the complexity of factors in the sharing and reuse of resources. In *Reusing Online Resources* ed. Littlejohn A. H., pp199-211. London, UK: Kogan Page