AC 2008-606: KEEPING TECHNOLOGY COURSES CURRENT WHILE MINIMIZING DISRUPTION TO THE INSTRUCTIONAL DESIGN

C. Richard Helps, Brigham Young University

Mark Patterson, University of Dayton

Keeping Technology Courses Current While Minimizing Disruption to the Instructional Design

Abstract

Technology changes rapidly which compels educators to rethink and redesign their courses. With Technology faculty being committed to experiential learning this implies the need for frequent redesign of technology courses, typically on a one to three year time scale. This paper discusses an integrated approach to instructional course design that provides the ability to adapt to changing technologies and incorporate current research into the curriculum. Learning can be improved and redesign costs minimized if there is a clear understanding of the relationship of the technical content to the overall instructional design. The instructional design presented here is conceived in terms of interacting layers analogous to Stewart Brand's architectural layers. Updating courses then becomes a process of changing the technical content layer while leaving most of the teaching material, organizational structure and learning objectives unaffected. This paradigm not only reduces the costs of updating courses but provides for a better collaborative student learning environment. Some aspects of this paradigm utilize reusable learning objects (RLO), student knowledge acquisition, and increased student participation to improve the learning and relevant instruction. In addition it provides faculty with more time to pursue other academic goals. This approach is discussed in the context of updating and re-designing technology courses at two campuses.

Introduction

Technology instruction should be based upon principles of educational interactions (teaching and learning) to change the student into a better learner, a competent technologist, and a life-long learner.

Learning in Technology emphasizes experiential learning. This has roots in constructionist and constructivist learning theories among others. This leads to a need to constantly re-design experiential learning experiences (labs and projects) to incorporate current technology to provide authentic experiences. The obstacles to achieving this in high-tech fields include the time and cost of constant re-design. In order to reduce the financial cost of redesign, faculty members sometime spend a lot of time seeking discounts and donors. In addition time and effort is expended negotiating with the administration for support for more new lab equipment. All of this leaves less time for teaching and research. Once the equipment is purchased courses and lab experiences are re-designed to incorporate the new technology. The technical turnover in technology classes is substantial. In electronics there is a continual drift towards more capable systems as they become cheaper, especially for digital systems driven by Moore's Law[1, 2]. If we do not use the latest technology is our classes, we are not serving our students (or our own scholarship) well. However, technology faculty share the common, painful experience of redesigning classes created by others or themselves based on a now-outdated textbook or lab equipment. The fact that academia generally doesn't recognize lab re-design or course re-design as scholarship for tenure and promotion exacerbates the problem. Thus there is a need for instructional design that at least addresses these problems. There are several possible approaches to this problem. One approach is to reduce the extent to which instructors employ current technology in teaching, concentrating on theory and principles in class instruction. This would reduce the value and authenticity of the experiential learning model. Another approach is to continue using experiential learning but minimize equipment changes in labs. This too jeopardizes the authenticity of the learning model, since students would either not be working with current equipment or would not be working with industry standard equipment. Alternatively the cost of equipment updates can be reduced though careful design, planning and interaction with vendors, donors etc. This option requires very large investments of faculty time. This paper will focus on an approach to instructional design of the course to incorporate new technologies with minimal effort while maintaining the instructional objectives and structure of the course.

Approach

In order to address the problem of continual course redesign we will start with the instructional design. A number of authors have discussed Re-usable Learning Objects (RLO)[3, 4, 5]. Robby Robson of Eduworks[6, 7] proposes an educational model based on re-usable educational objects which separate technology from pedagogy with respect to Content, Context, and Competency. He recommends the separation of the technical content, context and assessment from the learning intention. However, the authors think that while this approach is practical, it is a little too simplified. It addresses the need for content that can be shared in flexible ways but does not fully address the problem of rapidly changing content, often found in high-tech disciplines. Furthermore it does not address the problem of trying to maintain stability in instructional design and methodology while allowing content to vary. If we analyze what we are doing a little deeper it is possible to implement an instructional design based on Stewart Brand's architectural layers which separate the technical content, context, and assessment into separate layers. In the book "How Buildings Learn" Stewart Brand"[8] splits a building into several layers from the ground it sits on all the way through to the furniture that occupies the building. Each layer interacts with neighboring layers and is subject to different lifetimes. This same analogy can be and has been applied to instructional design. For example, the layers for a microcontroller class could be as follows. The technical *content* would be technology oriented (e.g. Microchip PIC architecture vs. Freescale architecture), the technical *organization* would be architecturally related (microcontrollers vs. conventional microprocessors, I/O issues/ Real-time issues, Operating system issues, etc.), and the *learning objectives* would be assessment based (technical competency, life-long-learning, communication skills etc.) as well as *several course management* and other aspects (how do you handle tests, record grades, evaluate course effectiveness, communicate with students, etc.). If we can successfully consider all these layers of course design separately then (hopefully) changing laboratory equipment suppliers from Microchip to Freescale, with the concomitant change of computer architectures, involves only changing one of several major aspects of the course, rather than re-designing the course again from scratch. Within each layer most RLOs would be kept and others would be created. Objects created at the *learning objectives* layer would most likely be similar from one iteration of the course to another and need little change. Objects at the *organization* layer would change slightly during a course redesign, but not significantly because the basic theories which govern our courses do not change that rapidly. Objects at the *content* layer would change significantly however. When a course redesign is accomplished, it makes sense to put the content specific issues at the lowest level (homework, classroom examples, laboratory exercises, etc) and leave the organization issues unchanged.

Observations

Knowledge acquisition and participation are the two prominent metaphors that guide our thinking about learning and relevant instruction. The first of them represents an individual and the latter a social basis of learning. Problem-based learning, case-based teaching, learning by design and cognitive apprenticeship powerfully emphasize anticipation and participation as main goals and perspectives into learning[9]. Those perspectives seem to be typical of all the approaches that can be related to the family of collaborative teaching models.

There is currently a lot of interest in the concept of learning objects. Learning objects are discrete units of learning resources based on agreed standards. The idea behind learning objects is to promote greater reuse of resources within new instructional systems development[10]. Learning objects can come from several sources. For example content layer learning objects come heavily from datasheets, while organization layer learning objects come primarily from textbooks and other publications.

Learning objects don't necessarily need to be acquired and developed by the instructor. One way to promote student life-long learning is to have them participate in finding and developing some of the learning objects and incorporating them into the course. In the book "Making Time, Making Change,"[11] Douglas Robertson shows several methods to reduce the overload on a teacher. One of these methods is have the students do part of the research to help keep a course current. Students in technology disciplines are often excited about cutting edge technology and it is thus easy to involve them in this learning experience. Robertson also employs "Non-Teacher Instructional Feedback," a technique to improve the quality of the students learning. This method can help to empower students to be self directed learners and also deepen their understanding of the learning process. One of his examples involves the students doing research on a certain subject and then presenting it to the class. Another method was quicker in that the student only had to explain it to another student at the beginning of class. One of the authors has found significant success in breaking the class into small groups and having each group very briefly research a topic and then explain it to each other or do one-minute presentations to the rest of the class. This alleviates some of the burden for the instructor for course updating and also enhances student learning. The students themselves are then providing the feedback that is needed for effective learning. Class participation is also often improved immediately after one of these exercises as students discover limits in their own understanding while they are trying to explain a topic to classmates.

There are several other methods to create learning objects. One method is to create a unified content strategy for learning materials[12]. However, a unified content strategy requires much up-front analysis. Implementing a unified content strategy involves deconstructing all content from all courses into elements. Another similar method is using a reflective group learning model to facilitate teaching[13]. However this involves three phases: an establishment phase, a preparation phase and class phase. The learning process involves rule setting, group formulation, individual study, group discussion, random group sharing, group presentation and individual reflection.

This method has been used in several learning environments. Another method from Liao suggests that students are able to achieve deep and meaningful learning through co-operative efforts. The motivation and involvement of the students enhances their learning. However, this method needs a systematic structure for the learning process and the continual encouragement of active participation in group discussion to ensure successful implementation.

An extension of the above concepts is the paradigm of Design Layers for instructional design. Design layers[14, 15] further develops the concepts suggested by Brand and others to suggest that instruction can be considered in terms of seven abstract layers which interact with each other. The layers proposed by these authors are: Content, Strategy, Control, Message, Representation, Media-Logic and Data-management. Here the abstract organization of the content is found within a single layer while many other aspects of the instructional design are viewed as separate layers. For example the instructor may choose a *strategy* based in constructivist, behavioral or other learning theories, *data management* could be handled using the course management software favored by the institution such as Blackboard or Moodle, *control* over the learning experience by the student could be implemented through classroom policies, and so on. With all this the design could remain intact while the domain-specific technical content within the *content* layer is allowed to vary. This system has been prototyped by one of the authors within a classroom environment and found to be effective both for student learning and for course stability[16].

Course Examples

Aspects of the techniques described above have been implemented by the authors. Following are several examples of courses that were designed or re-designed, together with issues and observations about those redesigns. In all of these cases an attempt was made to change the technical content while maintaining the pedagogical structure and management of the course.

The first example of a course redesign is for a programmable logic controllers (PLC) class. The PLC class was originally using Allen Bradley SLC-500 PLC's which are out of date. The laboratory component of the class was upgraded to the Allen Bradley Micrologix 1500 PLC's which luckily run on the same software and use almost identical ladder logic. This made the course redesign relatively painless. At the same time the textbook was changed from a book that only covered the SLC-500 to a text which covered almost every type of modern PLC with several additional items such as every type of programming language. Because the new text covered both the new and the old PLC's the conversion was relatively easy. New homework problems were chosen from the book, but the tests, course outline, and laboratory exercises remained largely unchanged. This same PLC class was also adapted to be taught in China in an outreach program with a different PLC architecture (Omron), which was locally available. Because the majority of the PowerPoint slides were organization layer information, they were able to be used with little modification even though their hardware was significantly different. The class examples were easily adapted to the new hardware because of similar functionality. The principles of PLC operation were the same. The reason that this course was relatively easy to redesign and update is that the instructional design layers worked properly. The learning objectives layer was left unchanged for the most part. The organization layer still maintained the same content (PLC architecture, scan cycles, Inputs/Outputs, State Machines, etc.), however there were some additions (programming languages, newer functionality, etc.) added towards the end of the course. The content layer in this class of the locally taught class luckily did not need major modification because of the similarity of the new hardware with the previous hardware.

Another example of a class that changes rapidly is a Telecommunications class. In this class, the instructor used student research, student papers, homework assignments, and student presentations to determine the most up to date information related to the class subjects. Much of the theory in Telecommunications stays the same, but the technology changes rapidly. If the instructor had to do all of the research for this class every year, it would swamp the instructor. The course was taught more like a theory class and the students found the recent, relevant technological information.

Another example of a class redesign was a mechatronics class that was split into three different classes. Because most of the reusable learning objects for this course were digital (presentations, homework, class examples, etc), they were fairly easy to add to the other courses.

The design layers paradigm was implemented in an operating systems class. In a similar manner to that recommended by Robertson[11], the students were invited to research specific topics on the leading edge of specific technologies, and share their findings with the rest of the class. The learning objectives, teaching strategies, teaching materials and many other aspects of the course were designed to remain constant although the specific technology changed.

A particularly challenging aspect of this design was the laboratory portion of the course. It has been found possible to make the instructional design accommodate technical content change by assigning the students to complete a project within the scope of the class, selected by the student and approved by the instructor, demonstrating accomplishment of class goals. Over the few years that this experience has been repeated the topics and scales of the projects have progressed steadily. For example a few years ago one of the advanced students selected as a very ambitious project implementing Myth TV, an open source programmable video recorder system, on a computer. This same project is now considered relatively straightforward and within the grasp of average students in the class.

There are, of course, some aspects of the lab equipment that require continuous refreshing. Specifically the institution has to continually seek to update its supply of available lab equipment so that students can pursue projects with reasonably up-to-date systems. However, by allowing the students to pursue individual projects the institution is freed from the need to have large numbers of identical lab configurations which they have to configure and maintain. The program updates its equipment supply by capturing "trickle-down" computers being re-cycled by the university as a whole, by recycling more capable machines originally purchased for research, by seeking donations through the industrial advisory board and by purchasing a relatively small number of new machines for the few projects that need them. The need for new equipment is also alleviated by the fact that a few enthusiastic students will provide their own project hardware each semester, to design a system that they wish to keep after the class is over. Thus while the need for some new equipment remains the constraints are significantly reduced and consequently the time and effort spent in acquiring new equipment is less.

Although the technical topics have advanced along with the technology, the assignment format, assessment rubrics and format and class management aspects have remained relatively constant, thus demonstrating that laboratory experiences can be evolved without extensive course re-design.

This project-based approach to experiential learning has worked effectively for up to 50 students a semester. Larger classes would be more challenging.

With certain classes one of the authors has had to do a wholesale change. An advanced microcontroller class was converted to use Field Programmable Gate Arrays (FPGA) rather than standard microcontrollers. Only about 1% of the instructional material could be re-used. In redesigning the class, material from a graduate EE class was reused. Some of the previous course reusable learning objects were moved to other courses and not lost completely. The content for the new FPGA course was designed with the new instructional design. The class content was designed with minimal references to specific hardware, but rather focused on the Verilog design language and hardware constructs. The laboratory was more hardware specific but by using a well-known FPGA development family it was possible to evolve relatively painlessly from the Xilinx Spartan II to the Xilinx Spartan III and Xilinx Virtex II.

In another microcontroller course, the laboratory equipment was upgraded from the Zilog Z80 processor to the Microchip PIC18. The architecture and instruction sets of the devices were significantly different; however there still were several similarities. Some concepts remain the same, for example LCD control and keypad interfacing. This class was also adapted for teaching in China with another microcontroller, the well-known 8051. During this course redesign, it was realized that even though they were using ancient technology, the fundamentals of the microprocessor class was much the same. Charles Reynolds[17], refers to this as "Pervasive Themes". While teaching in China, one of the authors realized that the core of microprocessor theory really hasn't changed too much over the years. There have been advances in capabilities, but the theoretical concepts are fairly constant. This realization reemphasized the need for this instructional design methodology.

Laboratories seem to be the most disrupted by technological changes. It takes a lot of time to change a lab setup and the costs are often high. In order to combat these costs one institution has recently designed more laboratory classes to be room independent. This did not reduce equipment costs but saved money by allowing fewer laboratory rooms to be used for multiple laboratory experiences. The institution also switched to a terminal server system this year and eliminated almost all desktop computers from the entire school of engineering to help minimize annual computer costs. Magazines such as Circuit Cellar have been incredibly helpful for learning about new development systems and getting them at a cheap price.

Conclusion

Every 3-5 years the technical content of the courses that we teach in higher education technology disciplines become obsolete. In order to maintain current with the latest technology, our courses have to be re-created. This leads to a need to constantly re-design experiential learning experiences to incorporate current technology. The problem with this process is that too many professors throw out all of the pedagogy material and much more along with the obsolete equipment. By considering the instructional design in terms of layers and separating the technical content, technical organization, and learning objectives from each other, only one aspect of the course may need to be updated and not everything. This technique applies to the laboratory as well as the classroom. In addition, by structuring projects and assignments as life-long learning experiences, the students can assist in this effort. This process will minimize the time required by the instructor to update the class and minimize costs with the changeover. With this process, we will also enable better learning through better instructional design - which design is likely to survive and improve through several cycles of updating the technical content.

BIBLIOGRAPHY

1. G. E. Moore (1965) Cramming more components onto integrated circuits, Electronics, Vol. 38, No. 8.
2. R. R. Schaller (1997) Moore's Law: Past, Present and Future, IEEE Spectrum, Vol. 34, No. 6, pp. 52-59.
3. D. A. Wiley (2001) Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy, in Wiley, D. A. The Instructional Use of Learning Objects: Association for Instructional Technology 2001.
4. M. D. Merrill (2001) Components of instruction: toward a theoretical tool for instructional design, Instructional Science, Vol. 29: 4&5, pp. 291 - 310.
5. R. McGreal (2004) Learning Objects: A Practical Definition, International Journal of Instructional Technology and Distance Learning, Vol. 1, No. 9, pp. 21-32.
6. R. Robson (2002) Reusable Learning Objects, in e-learning Magazine, Issue 9, pp. 18 - 19, 2002.
7. R. Robson (2004) Context and the Role of Standards in Increasing the Value of Learning Objects, in McGreal, R. Online Education Using Learning Objects (Open and Flexible Learning), pp. 159-167, Oxford, England: RoutledgeFalmer, 2004.
8. S. Brand (1994) How Buildings Learn: What Happens After They're Built. London: Viking Penguin.
9. J. Enkenberg (2001) Instructional Design and Emerging Teaching Models in Higher Education, Computers in Human Behavior, Vol. 17, pp. 495-506.
10. I. Douglas (2001) Instructional Design Based on Reusable Learning objects: Applying Lessons of Object-Oriented Software Engineering to Learning Systems Design, in Frontiers in Education, 2001.
11. D. R. Robertson (2003) Making Time, Making Change. Stillwater OK, USA: New Forums Press Inc.
12. P. Kostur (2002) Connecting Learners with Content: A Unified Content Strategy for Learning Materials, in ACM-SIGDOC Annual International Conference on Computer Documentation, pp. 100-103, 2002.
13. Z. Liao (1998) Reflective Group Learning Model for Case Studies in Engineering and Technology Management, International Journal of Continuing Engineering Education, Vol. 8, pp. 47-57.
14. A. S. Gibbons, and P. C. Rogers (2007) The Architecture of Instructional Theory, in 2007.
15. A. S. Gibbons, and P. C. Rogers (2006) Coming at Design from a Different Angle: Functional Design, in AECT Research Symposium, 2006.
16. C. R. G. Helps, and S. Renshaw, A. (2004) Design of a flexible case-study instructional module for operating systems for information technology, Proceedings of the 5th conference on Information technology education, pp. 56-59.
17. C. W. Reynolds (2006) Engineering the Information Technology Curriculum with Pervasive Themes, in.] Special Interest Group for Information Technology Education 2006 Conference, Minneapolis, Minnesota, USA: ACM, 2006.