Computers and Education

Computers and Education

E-Learning, From Theory to Practice

Edited by

Baltasar Fernández-Manjón
Universidad Complutense, Madrid, Spain

Juan Manuel Sánchez-Pérez
Universidad Extremadura, Cáceres, Spain

Juan Antonio Gómez-Pulido
Universidad Extremadura, Cáceres, Spain

Miguel Angel Vega-Rodríguez
Universidad Extremadura, Cáceres, Spain

and

José Bravo- Rodríguez
Universidad Castilla – La Mancha, Ciudad Real, Spain

Springer

A C.I.P. Catalogue record for this book is available from the Library of Congress.

ISBN 978-1-4020-4913-2 (HB)
ISBN 978-1-4020-4914-9 (e-book)

Published by Springer,
P.O. Box 17, 3300 AA Dordrecht, The Netherlands.

www.springer.com

Printed on acid-free paper

TABLE OF CONTENTS

PREFACE

Computers have long been used in education and today their use is having an ever greater impact on society thanks to the increasing educational success of the Internet and easier access to it. This broad field, which may include any kind of learning process aided by Information and Communication Technologies, is usually referred to as e-learning. E-learning is commonly recognized as a powerful and valuable extension to traditional educational initiatives and Learning Management Systems (LMS) are key tools that support these new educational models. For this reason a significant amount of research and development on both technological and educational issues in e-learning has been taking place with striking results. The field is beginning to come of age and is making important advances in the development, reusability and interoperability of educational content fostered by maturing standards and specifications (e.g. IMS, ADL/SCORM).

Nevertheless, even if we take these advances into account, more research and application work is needed in order to produce more cases of success and to generalize e-learning in industry, universities and schools. Certain problems have been identified such as those regarding costs, the knowledge needed to effectively apply this approach, or the fact that in many cases traditional LMS sometimes lack the required flexibility and adaptability to implement innovative educational models that need to be addressed. There are other issues also open such as how to involve learners further in the instructional process, how to stimulate the collaborative creation of educational contents, how to ease the creation and reuse of contents for non-experts, how to make more open and collaborative environments, or how to put into practice the effective integration of mobile devices in educational settings.

This book attempts to reflect several different views and efforts of the computers in education use. The book collects a set of selected and improved papers presented in the 4th International Symposium on Educational Informatics (SIIE in Spanish) hold in Cáceres (Spain) on November of 2004. The book includes also contributions of well known researchers in the

Educational Informatics field. The book chapters present experiences not only from the research point of view but from a practical point of view as well.

Cáceres, Spain, November, 2006

Baltasar Fernández-Manjón, UCM
Juan Manuel Sánchez Pérez, UEX
Juan Antonio Gómez Pulido, UEX
Miguel Ángel Vega Rodríguez, UEX
José Bravo Rodríguez, UCLM

ACKNOWLEDGEMENTS

The editors thank the Spanish Ministery of Science and Education for grant TSI2004-20279-E towards the edition of this book

Chapter 1

A HISTORY OF E-LEARNING

Echoes of the pioneers

Paul Nicholson
Faculty of Education, Deakin University, 221 Burwood Highway, Burwood, Australia 3125
pauln@deakin.edu.au

Abstract: In many contemporary sectors, E-learning is often regarded as a 'new' form of learning that uses the affordances of the Internet to deliver customized, often interactive, learning materials and programs to diverse local and distant communities of practice. This view, however, is historically disconnected from its antecedent instantiations, failing to recognize the extensive links between developing educational theories and practices that had shaped the use of E-learning over the past 40 years. In addition, the historic divide between Education and Training has led to both the concurrent development of different notions, foci, and labels for technology-enhanced learning in different contexts and situations, and different conceptual origins arising in acquisitive and participatory learning metaphors.

Key words: E-learning; history; theory; practice.

1. PARALLEL HISTORIES AND TERMINOLOGY

With the historian it is an article of faith that knowledge of the past is a key to understanding the present (Stampp in Szasz, 2006). In the history of E-learning, it is important to note that there is no single evolutionary tree and no single agreed definition of E-Learning: since the 1960s, E-learning has evolved in different ways in Business, Education, the Training sector, and the Military (for a military perspective see Fletcher & Rockway, 1986), and currently means quite different things in different sectors. In the school sector, 'E-Leaning' refers to the use of both software-based and online learning, whereas in Business, Higher-Education, the Military and Training sectors, it refers solely to a range of on-line practices. (Campbell, 2004)

B. Fernández-Manjón et al. (eds.), Computers and Education: E-learning, From Theory to Practice, 1–11.

The history of E-learning across all sectors is best summed up as: '*Opportunities multiply as they are seized*.' (Sun Tzu, 410bc) as for the past 40 years, educators and trainers at all levels of Education, Business, Training and the Military made use of computers in different ways to support and enhance teaching and learning. (Charp, 1997; Molnar, 1997) Consequently, the contemporary use of the term 'E-learning' has different meanings in different contexts (Campbell, 2004). In the Higher Education, Business, and Training sectors it relates particularly to Internet-based flexible delivery of content and programs that focus on sustaining particular communities of practice. E-learning in business and training can be characterised as being driven by notions of improved productivity and cost reduction, especially in an increasingly globalised business environment, with a focus on content delivery and online course management. These sectors initially employed the limited learning models extant at the time, but have since moved to incorporate a diverse range of learning models and foci. (Nicholson, 2004) Campbell (2004, p1) argues that:

> 'Broadly, in industry settings, E-learning reflects an emphasis on informal and non-formal, just-in-time learning where the emphasis is on collaborative productivity. Whilst, in higher education settings, best-practice online learning emphasises the development of metacognitive skills, where the emphasis is on reflective and collaborative learning.'

In the context of the wider education community, the use of the term E-learning has historically had wider connotations that embrace a diverse range of practices, technologies, and theoretical positions. It is not only focused on online contexts, and includes the full range of computer-based learning platforms and delivery methods, genres, formats and media such as multimedia, educational programming, simulations, games and the use of new media on fixed and mobile platforms across all discipline areas. It is often characterised by active learner-centred pedagogies. (e.g., Harel, 1991; McDougall & Betts, 1997)

The growth of E-learning in Business and Higher Education, and its marketing as a 'killer-app' (Friedman, 1999), has led to concerns about the influence of quality assurance driven models on the structure and quality of these programs (e.g., King, 2002; McGorry, 2003). Related concerns about its ability to deliver meaningful pedagogically structured learning experiences, or to have a clearly identifiable learning paradigm have also been raised (Gillham, 2002; Stone Wiske, Sick *et al.*, 2001; Suthers, Hundhausen *et al.*, 2003). Recently, driven by such concerns, its focus has expanded to accommodate the incorporation of learner engagement and social-learning models (e.g., Mortera-Gutiérrez, 2006; Schroeder & Spannagel, 2006). Since its inception, technological advances in computers

and networks facilitated advances in E-learning as educators seized on new features in an attempt to adapt them to their needs, to accommodate new educational theories, or looked for the promise of enhanced functionality. Curiously, many of these were foreseen by the pioneers of E-learning.

2. ORIGINS

The origins of E-learning as currently practiced in Business, Higher Education and the Military stem from the insightful work of Patrick Suppes at Stanford and Don Bitzer at the University of Illinois. While others such as Porter (1959) and Uttal (1962) were also active early in this field (Fletcher, 2002), only Suppes and Bitzer clearly situated the use of technology within a broader educational agenda (e.g., Suppes, 1964, 1966, 1986).

2.1 Patrick Suppes

In the 1960s, there were few educational applications of computers in universities, with most performing routine computational tasks. It was thought that the high cost of technology would prevent its ubiquitous uptake as an educational tool. In 1966 Suppes argued that '… in the future it would be possible for all students to have access to the service of a personal tutor in the same way that ancient royals were once served by individual tutors, but that this time the tutors would be in the form of a computer.' (Suppes, 1966). He argued that the single most powerful argument for the use of computers in education is individualized instruction and the dialogue that it supports. This was not an idle conjecture, but was based on Bloom's research that demonstrated that one-on-one tutoring improved student achievement by two standard deviations over group instruction – the equivalent of improving the performance of 50th percentile students to that of 98th percentile. (Bloom, 1984) Individual tutorials, Suppes argued, were also a core aspect of the university and computers would embrace and extend this through the use of virtual learning environments.

Driven by a belief in the educational potential of computers, Suppes founded the Computer Curriculum Corporation at Stanford as part of his ongoing inquiry into the nature, benefits and effectiveness of computer-enhanced learning. In accordance with prevailing psychological paradigms, he developed a Computer Managed Instruction system and used it widely in his courses. Suppes also provided elementary school children with individual CMI tutorials in mathematics to supplement teacher instruction. The results were inconclusive but led to suggestions for improved practices.

Suppes work and teaching was confined to structured fields and views of knowledge, with 'drill and practice' approaches being typical for such fields. He was concerned with both producing better learning, and learning how to be a better teacher with computers.

Contemporary critiques of his approach often overlook the lack of viable alternative paradigms at that time, something that Suppes was aware of. For example, in 1971 he noted that there was (then) a shallow understanding of how to use CAI effectively, and that it would take a long time to develop the necessary deep theoretical understandings that would underpin better practises. His research found that CMI produced profound effects on learning, and identified changes in students' understandings ranging from simple to complex. While his use of computers was essentially as a tool, he foresaw the potential for wider applications of computers in education. His research led to the following (amongst other) items for consideration:

- In 1971 the technology was not up to the tasks that he envisaged for it.
- The impediments to individual CAI were pedagogical not technological.
- CAI can track & follow each student, providing the potential for customised learning pathways.
- Richer learning theories were needed to inform design and practice.
- In the future, large numbers of students using CAI will be an important part of the mainstream university.
- There was a tendency to assess the product (of CAL) with simple studies using simple statistics when more complex measures might have led to more incisive conclusions.
- Students learning styles needed to be considered when developing CAL.
- How would more complex questions and responses be developed and handled as students increasingly engaged with higher-level content?

2.2 Don Bitzer: PLATO

In the early 1960s, Don Bitzer at the University of Illinois created PLATO, a timeshared computer system, to address concerns about student literacy. PLATO could be used to develop and deliver computer-based education, including literacy programs. It allowed educators and students to use high-resolution graphics terminals and an educational programming language, TUTOR, to create and interact with educational courseware and to communicate with other users by means of electronic notes – the forerunner of today's conferencing systems (Bitzer, Braunfeld et al., 1962). Woolley (1994) argues that as well as PLATO's advances in Computer Assisted Instruction, its communication features were equally innovative and were the foundations of today's conference and messaging systems:

> 'Two decades before the World Wide Web came on the scene, the PLATO system pioneered online forums and message boards, email, chat rooms, instant messaging, remote screen sharing, and multiplayer games, leading to the emergence of what was perhaps the world's first online community.' (Woolley, 1994)

When PLATO was eventually commercialised, it became the direct ancestor of today's E-learning systems such as Blackboard™ and WebCT™. It's interesting that what are widely touted as the key features of such systems are exactly those that Woolley identifies in PLATO! Like Suppes, Bitzer appears to have created the technology mainly as a tool, but also oversaw its operationalization in other dimensions.

3. FROM SIMPLICITY TO COMPLEXITY

When Dan Watt took the first computer terminal into a Boston school in 1969 , he could hardly have envisaged the subsequent changes that would occur – in particular the shift from localized 1:1 computing to distributed many: many models that occurred with the rise of constructivist and social-constructivist theories in the 1990s, and the related notions of situated and distributed cognition. To accommodate these cognitive and social learning theories required a major epistemological shift to embrace active learners, and indeed active communities of practice (Wenger, McDermott et al.), that were both knowledge consumers and knowledge creators (e.g., Papanikolaou, Grigoriadou et al., 2002).

3.1 Paradigm shifts

The eclectic history of E-learning means that constructs and paradigms in and across fields of use have merged and developed as part of the following trends in a progressive and incremental manner rather than being a new 'killer app' or 'a new way of learning'. The two interrelated trends examined briefly below (Figures 1 and 2 below) are the pedagogical focus of learning environments, and changes in the psychological foundations of learning. The size of the circles in those figures is meant to imply increased adoption or implementation over previous items, and is indicative only – they are not based on particular data. These meta-level characteristics of E-learning environments represent key lenses into what educators and developers were attempting to build and achieve with educational computing.

One of the most obvious trends in all areas of educational, business and training applications has been the increased scale of adoption of constructivist

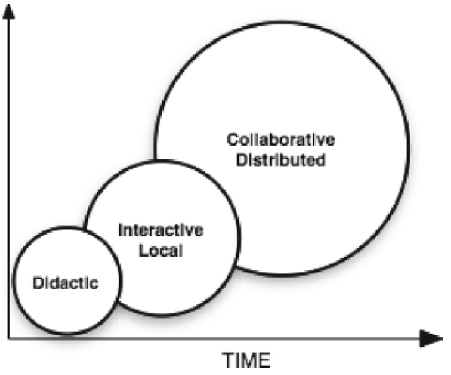

Figure 1-1. Trends in pedagogical stances over time. (Nicholson & McDougall, 2005)

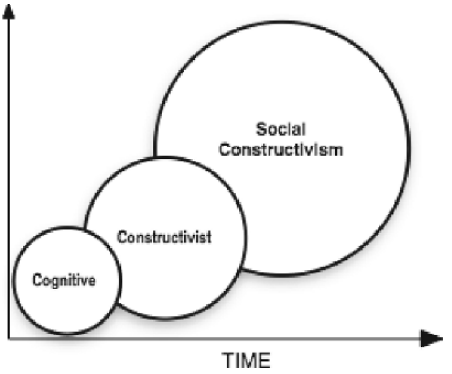

Figure 1-2. Development of learning paradigms over time. (Nicholson & McDougall, 2005)

paradigms, particularly social constructivism (Palincsar, 1998), distributed constructivism (Resnick, 1996), and the uptake of constructivist pedagogies (Forman, 1988; Ridgway & Passey, 1991). However, some care needs to be taken not to see this focus as being new or somehow being linked to the rise of ubiquitous networks etc. because its origins can be seen in both Bitzer's and Suppes' work, and because constructivist computing has been a key aspect of the use of computers in schools from the 1980s. However, in regard to the non-school sector, the constructivist trends shown below are arguably based more in the notions of communities of practice and computer-supported collaborative work (CSCW) than in constructivist psychology focus in schools.

3.2 Historical phases

Since its inception, E-learning has assimilated a diverse range of pedagogical practices, but the defining aspect of E-Learning—the trend towards collaborative online learning environments—is not only a result of the increasing adoption of constructivist paradigms, but is also a consequence of the affordances of ubiquitous global networks that have facilitated the realisation of individualised learning and interpersonal interactivity on a large scale, perhaps far exceeding the expectations of Suppes and Bitzer in its scale and scope.

3.3 A framework for comparison

Comparing E-learning practice over time is problematic and fraught with a host of methodological concerns. While Table 1 below provides an historical perspective based on macro-level features, it says little about the processes and agency occurring under the various categories. In order to make detailed comparisons of technology-based learning systems and paradigms over time it is necessary to explore complex interactions and

Table 1-1. The changing focus of educational technology over the past 30 years (after Charp, 1997; Herrington, Reeves et al., 2005; Leinonen, 2005; Mortera-Gutiérrez, 2006; Nicholson & McDougall, 2005; Pilla, Nakayama et al., 2006; THOMSON, 2005)

Era	*Focus*	*Educational characteristics*
1975-1985	Programming; Drill and practice; Computer-assisted learning – CAL.	Behaviourist approaches to learning and instruction; programming to build tools and solve problems; local user-computer interaction.
1983-1990	Computer-Based Training; Multimedia;	Use of older CAL models with interactive multimedia courseware; Passive learner models dominant; Constructivist influences begin to appear in educational software design and use.
1990-1995	Web-based Training	Internet-based content delivery; Active learner models developed; Constructivist perspectives common; Limited end-user interactions.
1995-2005	E-Learning	Internet-based flexible courseware deliver; increased interactivity; online multimedia courseware; Distributed constructivist and cognitivist models common; Remote user-user interactions.

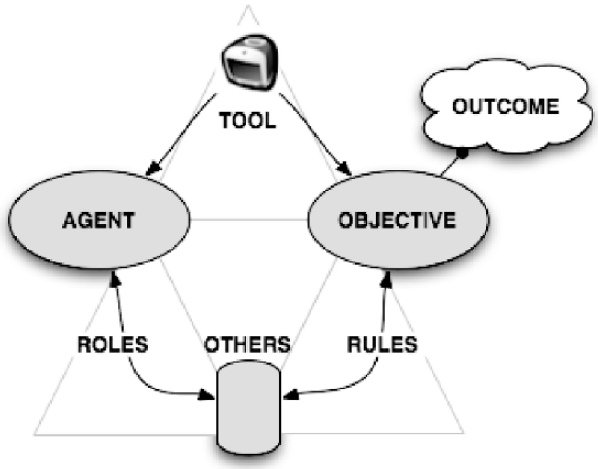

Figure 1-3. Activity Theory model for use with educational technology (after Nardi, 1996; Roschelle & Pea, 2002)

contexts. The nature of some of this complexity is revealed in Figure 3 – an adaptation of Nardi's Activity Theory model for use in technology-based learning contexts. The value of this model is that it provides a plausible, fruitful and comprehensible framework for use in exploring E-learning environments—as in Table 2 below. As an example of its use, Roschelle & Pea note that '…the tutor, tutee, tool debate (Taylor, 1980), …has largely focused on the topmost agent-tool-objective relationship of the diagram' (Roschelle & Pea, 2002, p.8). A focus on 'the others' and their couplings may include distributed and collaborative learning, social-constructivism and learning communities. Using agency as a probe makes it easier to examine and compare the nature of E-learning environments through better articulation of the nature and purpose of the Tool, its efficacy, and impact.

Table 1-2. Agency in Taylor's (1980) models of use. (Roschelle & Pea, 2002)

Model	Agent	Objective (goal)	Tool
Tool	Student	Semiotic tool	shared-knowledge
Tutor	Computer	student problem-solving behaviour	model-tracing
Tutee	Student	student-created program	microworlds

4. CONCLUSIONS

The contemporary claims for E-learning being 'new or different' arise in the different and independent development of the application of computers to educational needs in the business and education sectors, as well as from the 'lost history' of educational computing. It is clear that early pioneers such as Suppes and Bitzer, though confined by the dominant paradigms and technologies of their time, were striving to move beyond their contemporary practices to better engage learners and to enhance teaching and learning: at the inception of the field, PLATO contained features that pre-empted, and now characterise, cutting-edge third generation E-learning systems.

What we find by inspecting the past is that the notions of agency in contemporary E-learning systems can be found scattered heavily throughout past endeavours; the need to develop knowledge and skills, creating and interpreting texts (in the post-modern sense), synthesising and making sense of data, and creating new knowledge. Their model also highlights the need for such a model! It is difficult to begin to interrogate the diverse field of technology-enhanced learning without the aid of such a meta-level model.

While the hyperbole surrounding E-learning is not surprising given its origins in the business and training sectors, it is a concern that it has been so readily accepted as fact. While recent events have celebrated its rich history, the distillation of the 'lessons learned' and the developmental pathways has not been widely published or publicised.

The lesson of History has been that societies that don't understand their history are fated to repeat the mistakes of the past, suggesting that there is a need to make the history of the development of technology-enhanced learning more widely available and perhaps to consider its uptake as an element of professional development programs.

A focus on the Roschelle and Pea model may well be an important part of such programs as professional educators need to have appropriate tools with which to interrogate their field. Without such intellectual tools of inquiry to use in developing an historical perspective, we may well better appreciate William's caution…

History, history! We fools, what do we know or care?
(William Carlos Williams)

REFERENCES

Bitzer, D. L., Braunfeld, P. G., et al. (1962). PLATO II: A multiple-student, computer-controlled, automatic teaching device. In J. E. Coulson (Ed.), *Programmed learning and computer-based instruction* (pp. 205-216). New York: John Wiley.

Bloom, B. S. (1984). The 2-sigma problem: The search for methods of group instruction as effective a one-on-one tutoring. *Educational Researcher, 13*(6), 4-16.

Campbell, L. (2004). *What does the "e" stand for?* (Report). Melbourne: Department of Science and Mathematics Education. The University of Melbourne.

Charp, S. (1997). Some reflections. (the 30-year history of computers in education). *T H E Journal (Technological Horizons In Education), 24*(1), 8-11.

Fletcher, J. D. (2002). is it worth it? Some comments on reearch and technology in assessment and instruction. *Technology and Assessment: Thinking Ahead. Proceedings from a Workshop (2002)* Retrieved 07/07, 2006, from http://darwin.nap.edu/books/0309083206/html/26.html

Fletcher, J. D., & Rockway, M. R. (1986). Computer-based training in the military. In J. A. Ellis (Ed.), *Military contributions to instructional technology* (pp. 177-222). New York: Praeger.

Forman, G. (1988). *Constructivism in the Computer Age*. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Friedman, T. L. (1999, November 17). Next, It's E-ducation. *New York Times,* p. 25 (Section A).

Gillham, D. (2002). Web resource appraisal process (WRAP): A framework to establish critically appraised nursing knowledge--an active web based learning exercise. *Nurse Education in Practice, 2*(4), 257.

Harel, I. (Ed.). (1991). *Children designers: Interdisciplinary constructions for learning and knowing mathematics in a computer-rich school*. Norwood, NJ.: Ablex.

Herrington, J., Reeves, T., et al. (2005). Online Learning as Information Delivery: Digital Myopia. *Journal of Interactive Learning Research, 16*(4), 353-367.

King, K. P. (2002). Identifying success in online teacher education and professional development. *The Internet and Higher Education, 5*(3), 231.

Leinonen, T. (2005). (Critical) history of ICT in education - and where we are heading? [Electronic Version]. *FLOSSE Posse. Free, Libre and Open Source Software in Education*, 23 June. Retrieved 31-08-2006 from http://flosse.dicole.org.

McDougall, A., & Betts, J. (1997). *Learning with the media of their time: a snapshot of infrastructure, policy and practice in a technology immersion school*. Melbourne: Computing in Education Group of Victoria.

McGorry, S. Y. (2003). Measuring quality in online programs. *The Internet and Higher Education, 6*(2), 159-177.

Molnar, A. (1997). Computers in education: a brief history. *T H E Journal (Technological Horizons In Education), 24*(11), 63-69.

Mortera-Gutiérrez, F. (2006). Faculty Best Practices Using Blended Learning in E-Learning and Face-to-Face Instruction. *International Journal on E-Learning, 5*(3), 313-337.

Nardi, B. (Ed.). (1996). *Context and Consciousness: Activity Theory and Human-Computer Interaction*. Cambridge, MA: MIT Press.

Nicholson, P. S. (2004). E-Training or E-Learning? Towards a synthesis for the knowledge-era workplace. In P. S. Nicholson, Thompson, J., Ruhonen, M., Mulitsilta, J (Ed.), *elearning solutions for professional organizations* (pp. 360). New York: Kluwer.

Nicholson, P. S., & McDougall, A. (2005). eLearning: 40 Years of Evolution? In IFIP (Ed.), *The eighth IFIP World Conference on Computers in Education [ISI 1571-5736]*. Stellenbosch, ZA: IFIP.

Palincsar, A. S. (1998). Social Constructivist Perspectives on Teaching and Learning. *Annual Reviews of Psychology, 49*(1), 345-375.

Papanikolaou, K. A., Grigoriadou, M., et al. (2002). Towards new forms of knowledge communication: the adaptive dimension of a web-based learning environment. *Computers & Education, 39*(4), 333-360.

Pilla, B. S., Nakayama, M. K., et al. (in press). Characterising E-learning practices. In *Proceedings of WCC2002, Santiago, Chile, July 2006 [ISSN: 1571-5736]*. New York: Springer.

Porter, D. (1959). Some effects of year long teaching machine instruction. In E. Galanter (Ed.), *Automatic teaching: The state of the art* (pp. 85-90). New York: John Wiley.

Resnick, M. (1996). Distributed Constructionism. In D. C. Edelson & E. A. Domeshek (Eds.), *International Conference on the Learning Sciences: Proceedings of ICLS96, Evanston, IL July 25-27, 1996* (pp. 280-284). Charlottesville, VA: Association for the Advancement of Computing in Education.

Ridgway, J., & Passey, D. (1991). A Constructivist Approach to Educational Computing. *Australian Educational Computing,, 6*(2), 4-9.

Roschelle, J., & Pea, R. (2002, Jan 7-11). *A walk on the WILD side: How wireless handhelds may change CSCL.* Paper presented at the CSCL2002: Computer Supported Collaborative Work, Boulder, Colorado.

Schroeder, U., & Spannagel, C. (2006). Supporting the Active Learning Process. *International Journal on E-Learning, 5*(2), 245-264.

Stone Wiske, M., Sick, M., et al. (2001). New technologies to support teaching for understanding. *International Journal of Educational Research, 35*(5), 483.

Sun Tzu. (410bc). *The Art of War*.

Suppes, P. (1964). Modern learning theory and the elementary-school curriculum. *American Educational Research Journal, 1*, 79-93.

Suppes, P. (1966). The uses of computers in education. *Scientific American, 215*(206-220).

Suppes, P. (1986). Computers and education in the 21st century. In W. Neilson & C. Gaffield (Eds.), *Universities in Crisis: A mediaeval institution in the twenty-first century* (pp. 137-151). Toronto: The Institute for Research on Public Policy.

Suthers, D. D., Hundhausen, C. D., et al. (2003). Comparing the roles of representations in face-to-face and online computer supported collaborative learning. *Computers & Education, 41*(4), 335.

Szasz, F. M. (2006). Quotes about History. from http://hnn.us/articles/1328.html

Taylor, R. (Ed.). (1980). *The computer in the school: Tutor, tool, tutee*. NY: Teacher's College Press.

THOMSON. (2005). History of E-Learning. from http://www.knowledgenet.com/corporateinformation/ourhistory/history.jsp

Uttal, W. R. (1962). On conversational interaction. In J. E. Coulson (Ed.), *Programmed learning andcomputer-based instruction* (pp. 171-190). New York: John Wiley.

Wenger, McDermott, et al. *Cultivating Communities of Practise: A guide to Managing Knowledge*.

Woolley, D. R. (1994). PLATO: The Emergence of Online Community. Retrieved 30/12, 2004, from http://www.thinkofit.com/plato/dwplato.htm

Chapter 2

.LRN: E-LEARNING INSIDE AND OUTSIDE THE CLASSROOM

Supporting Collaborative Learning Communities using a Web Application Toolkit

Carl Robert Blesius[1], Pablo Moreno-Ger[1], Gustaf Neumann[2], Emmanuelle Raffenne[3], Jesús González Boticario[3], Carlos Delgado Kloos[4]

[1]Harvard Medical School - Massachusetts General Hospital, Laboratory of Computer Science, 50 Staniford St. MA 02114 (USA); [2]Vienna University of Economics and Business Administration (Austria); [3]Universidad Nacional de Educación a Distancia-UNED (Spain); [4]Universidad Carlos III de Madrid (Spain)

Abstract: .LRN is an Open source Web portal and Web application toolkit designed to support both large and small communities of practice and learning inside and outside of educational institutions and the enterprise. .LRN has the features of a complete Course Management System, but is focused on supporting collaborative online learning communities that often go beyond the typical institutional or course based setting. This chapter describes the main features of .LRN, gives an overview of its modular and adaptable technical architecture, includes case studies from projects the authors are directly involved with (which help highlight the benefits of the platforms open source nature and its high performance capabilities), and concludes by giving a quick summary of the future direction of .LRN and the not-for-profit corporation that has formed around the software.

Key words: Collaborative Learning, .LRN, OpenACS, LMS, CMS, Web Portals, Software Consortia, Open Source, E-Learning.

1. WHAT IS .LRN?

.LRN is a fully internationalized open source portal and application framework built to support online collaborative learning communities and blended learning environments. .LRN is based on the tenet that learning is a

B. Fernández-Manjón et al. (eds.), Computers and Education: E-learning, From Theory to Practice, 13–25.

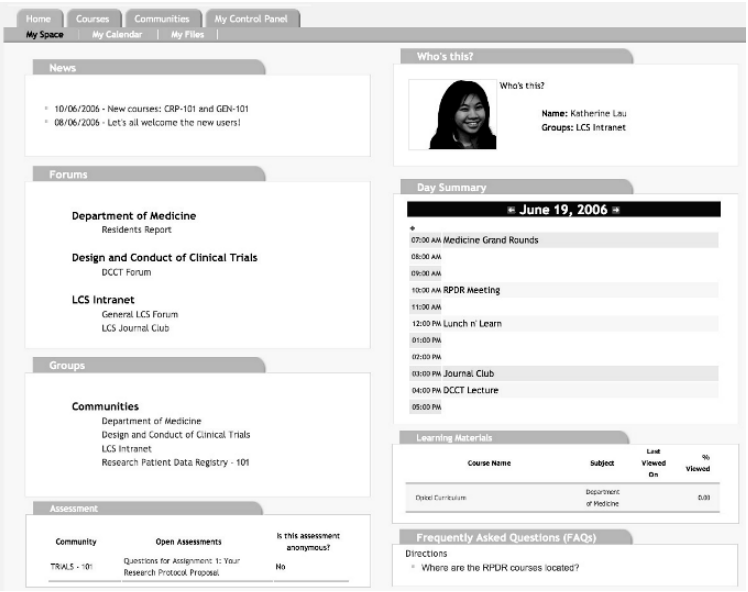

Figure 2-1. A screenshot of the .LRN personal portal page. It aggregates information from community and class portal pages. End users can customize this page as they please

social experience (Wegerif 1998), that effective learning usually takes place in the context of communities (Alavi 1994; Wenger 1998), and that administration of these communities should be distributed. Each learning community has its own stakeholders and needs to define its own unique set of interactions, so the software is designed to be flexible and allow delegation of administrative roles as close to the learners as possible. While some e-learning environments are built around a course catalog (course management systems, CMS) and other are built around a content management system (learning content management systems, LCMS) .LRN focuses on online communities (learning community system, LCS), with course management and content management applications as an added value. This contrasts with the more content-centric approaches of most of the key systems in the e-learning arena, like the commercial WebCT (Goldberg and Salari 1997) or the Sakai (Farmer and Dolphin 2005) opensource project and taking the community oriented approach of Moodle (Dougiamas and Taylor 2003) to the extreme.

From an e-learning perspective, some interesting samples of .LRN applications are the Learning Object Repository System (LORS) that supports the IMS Content Packaging Specification (allowing packaging, organization, and import/export of learning objects), the Assessment tool (tests, quizzes, surveys, evaluations, etc.) that supports the IMS QTI import

Table 2-1. A rough order of magnitude estimation of the .LRN development cost based on lines of source code

Total Physical Source Lines of Code (SLOC)	487,182
Development Effort Estimate, Person-Years (Person-Months)❶	132.77 (1,593.26)
Schedule Estimate, Years (Months) ❷	3.43 (41.19)
Estimated Average Number of Developers (Effort/Schedule)	38.68
Total Estimated Cost to Develop (average salary = $56,286/year, overhead = 2.40).	**$17,935,662**
❶Basic COCOMO model, Person-Months = 2.4 * (KSLOC**1.05)	
❷Basic COCOMO model, Months = 2.5 * (person-months**0.38)	
Data generated using David A. Wheeler's 'SLOCCount'	

and export, Gradebook, Homework (an assignment drop box for course facilitators), User Tracking (allows administrators to track what users have or have not seen), the ePortfolio package (allows individuals to keep track of their personal learning progress and share their interests and accomplishments with others) and, optionally, E-Commerce support (e.g. credit card based course registration).

.LRN is about learning communities and these are supported by applications such as the Forums and News applications (with email and RSS subscription options), a WYSIWYG Wiki system (which can be used in the context of the .LRN Learning Object Repository or as a simple content management system), Photo Album, group and individual calendaring (with Outlook synchronization), Weblogger, and Web-based group/personal file storage with WebDAV support.

The first version of what would become .LRN was created by ArsDigita Corporation (which has been sold to RedHat) for the Sloan School of Management (MIT's business school) and was based on software ArsDigita built for corporate knowledge management, intranets, and dynamic database backed websites. The system that was created for Sloan was called the *"ArsDigita Community and Educational Solution (ACES)"* (Meeks and Mangel 2000) and was released under the GNU Public License. The system was built to run on Oracle, but in 2001 a group of volunteers started the work required to support the PostgreSQL open source database.

Simplistic development cost calculations (Table 2-1) give a very rough estimation of the amount of development (~133 person years) and investment (~$18,000,000) that has gone into the software.

2. TECHNICAL BACKGROUND

.LRN is built using OpenACS (Calvo and Peterson 2002; Hernández 2005), a mature enterprise toolkit for building scalable web applications. OpenACS follows a multi-tier software design pattern (Web server,

application server, and database) and depends on AOLserver (Reuven 2002) (an application/Web server), which was built from the ground up as a tool for building highly scalable database-backed web applications.

2.1 General architecture

Based on almost 10 years of experience, the .LRN platform has evolved into a modular web application, with the goal of providing a sustainable and rich architecture, which promotes extensibility and adaptability without sacrificing maintainability. This is achieved with an organization in which functionality is divided into *packages* that can be designed, maintained, and distributed independently.

A default installation of the platform includes the core packages that provide the infrastructure of the system, a portal infrastructure (Fig. 2.1) and the basic set of .LRN packages (Fig. 2.2) necessary to support a typical E-Learning scenario, however there are dozens of official packages that can be added to the default installation. In addition, a mature package development process allows numerous non-official packages to be created by third parties. This allows institutions to create their own additions to the basic OpenACS/.LRN installation and encourages them to share their creations. This is similar to the Building Blocks program around Blackboard's open architecture (Blackboard Inc 2006), although the distribution of the packages as open source applications is more common in the OpenACS/.LRN community.

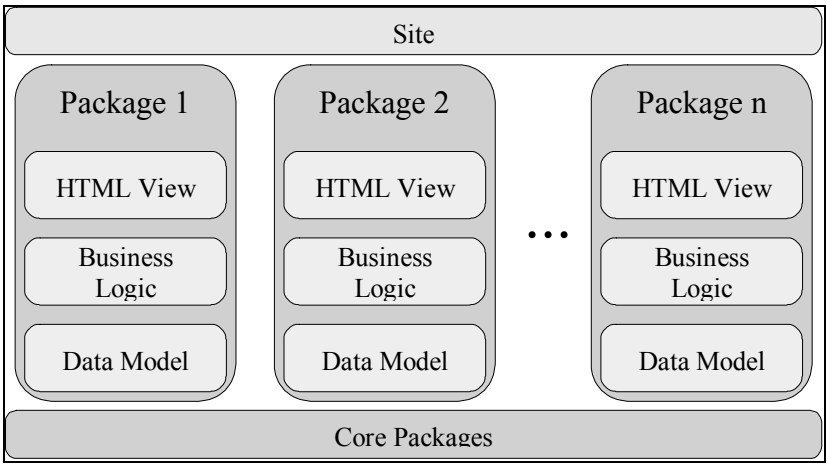

Figure 2-2. General architecture of the OpenACS/.LRN platform. The "site" layer is a common entry point for all the packages, where the look and feel of the site is defined.

Generally packages are designed following a Model-View-Controller pattern that separates the view (HTML syntax), the business logic (Tcl scripts), and the data model. Typically the resulting packages are web-based software applications that take advantage of base functionality provided by core packages and can be distributed independently.

2.2 Separation of the View and Business Logic: The Templating system

Decoupling the view from the functionality itself provides a number of advantages in the development of web applications, most of them derived from the fact that neither web programming nor HTML design are trivial tasks and that both require a different set of skills. In fact, the first benefit obtained from this separation is the definition of two separate roles: The web *programmer* and the web *designer*.

The .LRN platform provides such a separation by means of the Templating System. Every request that the server receives is handled in two steps. First, a Tcl script (Ousterhout 1994) is executed. This contains the business required to attend to the request, like checking for permissions, accessing a database, modifying some data or preparing some data for display. The actual rendering of the page is postponed until the second step, the execution of an ADP page (AOLserver Dynamic Page), which is a simple document containing HTML markup and special tags. The key idea is that creating an ADP page does not require special programming skills (other than HTML).

2.3 Implementation of the Business Logic: The Application Server and the Web Application Programming Interface

The evolution of the .LRN platform and its backing software (namely, OpenACS and AOLserver) provides a number of programming APIs that facilitate the implementation of the business logic layer with different level of abstraction. At the core of the system, AOLserver provides a highly efficient multithreaded back-end, which is based on resource pooling and a caching mechanism, with a throughput capable of sustaining sites such as www.aol.com and www.mapquest.com. AOLserver provides a fully featured API for application programmers that can be called using Tcl.

However, it is not common to have to deal with this low-level API, as .LRN provides a rich API that facilitates development with high-level