Teaching Systems Analysis and Design Problem Solving with Interactive Multimedia and Patterns:

White Paper

Jacob L. Cybulski

Department of Information Systems
The University of Melbourne
Parkville, Vic 3052, Australia
Phone: +613 9344 9244, Fax: +613 9349 4596
Email: j.cybulski@dis.unimelb.edu.au
Internet: http://www.dis.unimelb.edu.au/staff/jacob/

Abstract

This paper describes the on-going project aiming at the development of an intelligent multimedia system assisting university instructors in the effective teaching of systems analysis and design skills to Information Systems students. Our tool will comprise a multimedia library of problem-solving patterns and a decision-support system matching student problems against a set of possible problem solutions. Retrieved solutions will be presented to students as a collection of self-paced web and CD-ROM based mini-lectures, examples and tutorials, coupled with the interactive and self-assessing exercises and projects. In the future, the problems and their solutions will span several DIS subjects across different years of study, they will be cross-referenced and linked for easier navigation between different levels of study and understanding, teaching units and problem solving components.

1. Introduction

Multimedia (i.e. the ability to combine and simultaneously present text, graphics, animation, sound, video and the output of executable programs) can add significantly to the value of some knowledge-intensive services such as computer-assisted education and training, electronic publishing or telemarketing [1, 8, 11, 37].

In particular, in computer-based education, multimedia was found to improve students learning [4, 12, 15, 25, 29, 40] and understanding [22], enable development of their interests [3, 43] and motivation [41]. Even the simplest multimedia presentations, such as those created with Microsoft PowerPoint [39, 44], significantly increase the effectiveness of traditional teaching methods with the use of colour, sound and movement [5]. In general, such presentation mechanisms are more suitable to the lecture-style delivery of information, which is non-interactive, synchronous, face-to-face, en-masse, and which puts the teacher in control of the pace, contents and the flow of information. Interactive multimedia, on the other hand, is suited to the tutorial- or laboratory-style of presentation [2, 14, 21, 33, 44]. It provides students with more control over contents, sequence and flow of educational material [18, 19, 24, 32, 38], pace of learning [30, 35] and the amount of received feedback [31, 36, 42].

Multimedia provides students with the ability to absorb educational material in a variety of media. This not only improves its aesthetics, but also presents concepts in a variety of contexts and from a variety of perspectives. “Simulations, immersion learning, visually rich case studies and showing things, people and processes from different times and different places just can’t be

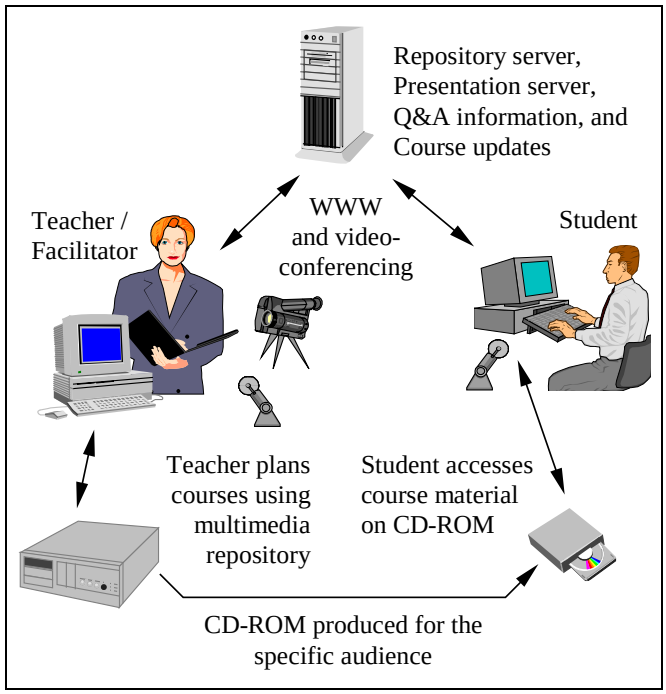

Figure 1 - Teacher's MATE architecture

done without multimedia. There is a texture, a richness, that can't be achieved in any other way." [13]. It is these characteristics that make complex concepts easier to understand with multimedia in a way that is not possible with static imagery.

Experimental studies show that multimedia has a significant positive effect on students' academic achievement [23], their attitude towards the subject matter [27], and their perceived satisfaction with the learning experience [26]. This satisfaction has been found to be a key factor in the positive attitudes by students toward the new technology [6]. Students appear to be more satisfied because of the increased interactivity, reduced learning time, high achievement and the acquisition of new technical skills [7]. All of these benefits are increasingly evident when teachers also have high expectations of the computer-based instruction, which in turn favourably affects students' attitudes, interests and performance [23].

At the same time, however, development of multimedia presentation material requires complex software tools, as it is extremely laborious, and very expensive [34]. For certain groups of users, such as school teachers or university lecturers, multimedia technology is quite inaccessible - the process is too technologically involved, too lengthy, and the cost of employing a multimedia professional is, needless to say, prohibitive.

We believe that the productivity of developing multimedia material can be improved and the costs can be reduced with the introduction of component technology and reuse techniques, both being vigorously researched in the area of software engineering [9, 10]. Commonly reused software artefacts include binary programs, their source code, diagrammatic designs and formal specifications. All such artefacts are produced according to well-established methodologies, they are usually well structured, having well defined formal syntax and semantics, they are complete and adequately documented. Multimedia artefacts are quite different - they are created in an ad-hoc fashion, they are ill structured, informal, with no rigid syntax or semantics, frequently incomplete and most often not documented in any way [28]. In this situation, few researchers have moved beyond reusability of clipart or the use of web search engines in the pursuit of multimedia artefacts. Practical reuse of multimedia components is a matter of urgency!

2. Teacher's MATE

We have undertaken the challenge of effective reuse of multimedia components in our Teacher's MATE (Multimedia-Assisted Teaching Environment) project. In the project, we proposed several techniques suitable for the analysis and storage of multimedia components and the composition of new presentations of reusable multimedia artefacts. We address the entire cycle of multimedia reuse and propose necessary methods for the efficient processing of multimedia information, i.e. its identification, representation, generalisation, classification, storage, search, retrieval, selection, composition and integration. We are also in the process of formalising reusability techniques for a variety of basic multimedia components (such as text, graphics, sound, animation and video),

structured components, presentation sequences and the methods of their generation. In doing so, we will ultimately develop an integrated technological framework for dealing with multimedia documents and their components.

As part of the Teacher's MATE project, we developed a prototype tool that allows teachers to record, organise and cross-reference sound, video, animation, text and graphics. The system also helps teachers in the planning and delivery of fully interactive lectures and tutorials to a large number of simultaneous student users on the web. With the aid of Teacher's MATE, students are able to work alone (off-line or on-line), in small teams, or under the direct instruction and supervision of their teacher. Teachers are also more effective as they are equipped with tools capable of browsing, searching, retrieving and presenting vast volumes of teaching material stored locally on student machines and the web.

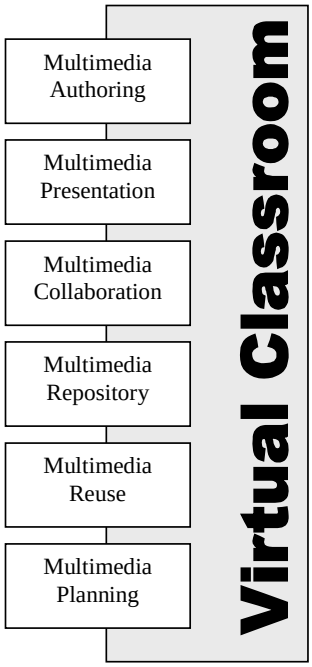

Figure 2 - Virtual Classroom

In a typical course of events (Cf. Figure 1), teachers produce multimedia material using existing web-compatible authoring tools (e.g. drawing and painting packages, HTML editors, sound and video recording software, animation packages). Each multimedia artefact is then classified and subsequently added to the Teacher's MATE repository of multimedia components. Having a rich collection of reusable components, teachers use them to compose and structure multimedia slides, lectures and courses. A course can then be exported to CD-ROM in a form compatible with a number of commercially available web browsers, e.g. Netscape Navigator or Microsoft Internet Explorer. Students can then browse CD-ROM material off-line or they can attend a real-life on-line lecture. During the presentation, the teacher uses a lecture plan to identify the lecture contents, default sequence of multimedia slides, alternative lecture scenarios, slides useful to answer students' ad-hoc queries, and other freely available web resources. Lecturers can then request the remote display of lecture components stored on the student's CD-ROM or the web. At any time after the lecture, students can also re-trace and view already presented lecture slides.

Having the capabilities to effectively manage multimedia authoring, presentation, collaboration, storage, reuse and planning, Teacher's MATE facilitates the organisation of a *virtual classroom* on the World Wide Web (Cf. Figure 2).

We are also planning to use Teacher's MATE as a vehicle for the creation of a system that will guide IS students in the process of solving typical software development problems.

3. Teaching IS with Multimedia and Patterns

Practice shows that the most difficult tasks to both IS students and IS practitioners are those related to the problem analysis, solution and evaluation rather than the tasks leading to the implementation of the working computer programs.

The main difference between novices and experts in the IS field, is the ability of the latter to form sophisticated mental schemata useful in categorising problems in the domain, identification of the problem features and the selection of the most appropriate problem solutions [20]. Recent advances in software design adopt the cognitive view of the IS problem solving and propose the use of the so-called "analysis and design patterns" [16, 17]. A large collection of such patterns can be used and combined to form high-quality software designs.

Modern teaching of Systems Analysis and Design recognises the value of pattern-based software development. Problem-solving patterns are discussed in lectures, tutorials and laboratories. Students learn the most commonly used patterns in their project work and during consultations with their project supervisors. However, due to a very large number of useful

patterns, neither students nor the less experienced members of staff are able to learn or apply patterns effectively.

This project proposes to build a large repository of IS problem-solving patterns and their examples, presented in the multimedia form, and aided with a tool capable of guiding students towards a high-quality problem solution. Retrieved solutions will be presented to students as a collection of self-paced and teacher-assisted multimedia lectures, examples and tutorials, coupled with the interactive and self-assessing exercises and projects. The problems and their solutions will span several DIS subjects across different years of study, they will be cross-referenced and linked for easier navigation between different levels of study and understanding, teaching units and problem solving components, e.g.

- Information Systems in Organisations (300 students)
- Introduction to Programming (300 students)
- Object-Oriented Programming and Design (200-300 students)
- Systems Analysis and Design (100-200 students)
- Requirements Engineering (40 students)

Such a repository and a tool will not only help students with their project work but will also facilitate learning of useful problem-solving patterns, thus, making IS students self-reliant, more confident and more independent.

Problem: A typical approach to solving an IS problem includes the following tasks:

1. analysis of the problem at hand;
2. finding, selecting and applying the solutions appropriate for the problem;
3. encoding the selected solution into an executable program;
4. testing the programs; and finally,
5. evaluating the success of the problem-solving activities.

Aim: To better facilitate acquisition of Information Systems (IS) problem-solving skills by IS students, we proposed to develop:

> *an intelligent, multimedia tool that could assist second year students in studying and applying methods and techniques used in the process of solving typical analysis and design problems.*

Objectives: In the process of developing the tool, we are also hoping to reach the following objectives as well:

- To make explicit what to this date was hidden, implicit, and was assumed to be common sense but what IS students frequently find difficult.
- To identify and describe a wide range of activities performed in the process of solving a typical IS problem.
- To organise the techniques associated with IS problem solving activities into a collection of reusable context-problem-solution-consequence patterns.
- To develop a series of interactive, multimedia, self-paced mini-lectures, examples and exercises teaching problem-solving with IS patterns.
- To integrate IS patterns with the rest of DIS curriculum.
- To implement a decision-support system that could advise students on the best selection of commonly used solutions to a given IS problem.
- To assess effectiveness of teaching DIS curriculum with the use of IS pattern database.

4. Sample Problem-Solving Patterns

Note that all seemingly "sophisticated" methods requiring theoretical or psychological underpinning will be turned into ready-made, self-guided, intuitive and simple problem solutions. The emphasis will be on the practical application of any of the IS patterns rather than on the explanation of its proof-theoretical basis. The following table provides a sample of list of useful problem-solving patterns.

Problem solving activity	**Sample Patterns, Subjects and Issues**
1. Prerequisites to problem solving	*To be an effective problem solver you need to learn more about your own mind, its workings and its limitations*
• Memory	features, chunks, patterns, probability
• Learning	loci, peg words, context, PQ4R, reconstruction
• Perception and attention	divided attention, automaticity, template matching, feature analysis, Gestalt, familiarity, context
• Perception-based representation	mental pictures, lists, maps, ordering, transformations and mental imagery
• Meaning-based representation	propositional representation, semantic networks, schemas and frames
• Problem solving	problem states, GPS, difference-reduction, means-ends analysis, working backwards, analogy
• Reasoning	propositional logic, heuristics, probability and possibility, hypothesis formation and evaluation
• Language	comprehension and accuracy, listening and speaking, memory, reading and writing
2. Collaborative problem solving	*IS problems usually require a team approach to problem solving, there are many well-known approaches to facilitate better group communication and problem solving*
• One-one communication	chat, explanation, interviews, feedback
• One-many communication	presentation, eye contact, attention control
• Many-many communication	brainstorming, discussion, debating, controlled meetings, argumentation, agenda, minutes
3. Information acquisition and problem identification	*Of equal importance to problem solving is to be able to identify the problem in the first place and then to describe it in detail*
• Information gathering	interviews, observation, surveys, meetings and discussions
• Information identification	verb-noun analysis, protocol analysis, process tracing, concurrent and retrospective verbalisation, discussion, cued recall, scenarios and simulations
• Information organisation	comparing and contrasting, generalisations, predictions, conceptual dictionaries and frameworks, cognitive maps and models, taxonomies, concept sorting, scaling and clustering, repertory grids, affinity analysis
• Ranking of collected information	analytic hierarchy, pairwise comparison
• Identifying opportunities	future perfect, CRLC, PIECES, CSF, 5W1H, user context
• Risk analysis	risk evaluation form, what-if-analysis, capacity planning, forecasting, simulation
• Establish goals and problem description	templates, forms, outlines, diagrams

Problem solving activity	Sample Patterns, Subjects and Issues
4. Problem analysis	
• Abstraction	abstract data types, feature and behaviour identification, abstracting data and functions
• Representation	data structures: numbers, array, record, file, propositional, rule-based, hierarchical, correspondence and equivalence
• Problem decomposition	refinement, transform and transaction analysis, rules for stopping, normalisation
• Generalisation and specialisation	clustering, substitution, relaxation, integration
• Categorisation and classification	typicality, similarity and variability, enumerative, faceted and attribute-value classification
• Unifying multiple views	brainstorming, debriefing, group consultation, consensus decision making, nominal group technique, multiple lines of reasoning, meta-rules, fuzzy sets, uncertainties, CORE, JAD
5. Problem solving	
• Search and planning	depth-first, breadth-first, best-first, beam-search, divide-and-conquer, hill-climbing, mini-max, A*, alpha-beta
• Deduction and induction	various reasoning heuristics, forward-chaining, backward-chaining, and-or trees
• Heuristic solutions and abduction	pruning, rules-of-thumb
• Restructuring, insight and analogy	case-based reasoning, analogical reasoning
• Parallel constraint satisfaction	constraints propagation
• Probability and possibility	fuzzy logic, rough sets
6. Solution evaluation and selection	
• Belief and judgement	bayesian competence, similarity, odds, conjunction, coexistence, randomness
• Choice and decision making	maximisation, sure-thing principle, utility, satisficing, preferences, gambling
7. Solution design and implementation	*Adopted from the Gang of Four book but many more coding-oriented patterns may be added*
• Conditional computation	responsibility chain, factory, builder (case, if, selection)
• Iteration and recursion	iterator (loops, invariance, recursion types)
• State	state (global, information hiding)
• Parametrised procedures	command (procedure call)
• Module interfacing	adapter, facade, bridge, decorator, mediator, proxy (client-server)
• Program composition	abstract factory, composite, observer, visitor (control, communication, message passing)
• Aggregation	flyweight, strategy, template (arrays, lists, files)
• Interpreted programs	interpreter (table-driven processing, STD)
• Planning for change and reuse	macro, class, module, template, data dictionary, library
8. Solution validation and verification	
• Experiments and tests	e1quivalence classes, boundary-value analysis, cause-effect analysis, comparison testing, basis path testing, condition testing, data flow testing, loop testing, mutation testing
• Proofs of correctness	invariance testing, algebraic proofs, model proofs
• Software reliability	Reliability and availability metrics, safety models
• Quality assurance	reporting, walkthroughs, technical reviews, cleanroom process, statistical methods

5. Evaluation of the Project Success

Expected Benefits: The system will have positive effects in three areas, i.e. our teaching methods, student learning abilities and staff workload.

The pattern-oriented approach to problem-solving will affect our teaching methods, it will:

- complement tutorials and labs with a rich collection of mini-cases, precedents and problem-solving patterns;
- make learning of IS skills more explicit and the process of learning these skills easier to control;
- constitute a common point of reference across the second and third year of IS curriculum, which will facilitate consistency between different subjects and the staff involved in shared teaching.

The proposed system, will enhance students learning by allowing them to:

- compare and contrast different approaches to problem-solving by navigating the pattern library;
- learn solutions to problems from different vantage points that will depend on the student knowledge, the problem area, and the methods and tools to be used;
- use simple patterns directly while solving simple problems or to combine them in larger projects;
- rely on their own abilities to cope with the problems they face and become less dependent on the help of their teachers - this will not only reduce the workload of IS educators but will also make our students more confident and more competent problem solvers.

The system should have a positive effect on staff and student workloads:

- our 2nd and 3rd year teaching staff will be able to incorporate and reference reusable teaching material in their lectures and tutorials, hence, the effort of preparing the new teaching material in multimedia form will be greatly reduced;
- we expect that due to students' enhanced ability to solve IS problems on their own, the staff-student consultation time will also be decreased;
- students workloads are not expected to be increased in any way. Students will be less dependent on the availability of staff for problem-solving consultations.

Evaluation: We will evaluate the success of our method (in the identified three areas) over two years (1998-1999) as follows:

- we will design a set of measures and monitors (manual and automatic) to collect data on student and staff effectiveness in problem-solving and in the use of the tool (1998);
- we will monitor student and staff use of the tool to determine whether or not we achieved the expected benefits (1999);
- we will conduct a survey of students' satisfaction with the use of the pattern tool (1999);
- we will conduct a survey of staff satisfaction with the use of the tool in the preparation of their multimedia lectures (1999);
- the effectiveness of the new method of IS teaching will be continually evaluated after 1999.

6. Proposed project schedule

		Deliverables on Deadline	
Deadline	**Project Reporting**	**Tasks performed by Chief Investigator**	**Task performed by Students / Assistant**
1998 January		Development of a sample multimedia repository	Selected software tools purchased
February			Development of the multimedia storage system
March			
April	Project white paper	System design	Development of the prototype (Excel) for the UML reuse assistant
May	Patterns in software requirements	Development of electronic story board	
June	Reuse of software requirements		
July	Multimedia patterns		
August	UML patterns & reuse	Data collection	Development of SA&D patterns (as published by Gamma and Larman)
September	Patterns in SA&D education		
October		Experiments in use of the prototype by students	Two fractional research assistants hired
November	Use of multimedia, patterns and reuse in teaching SA&D	Completion of the SA&D and multimedia pattern language	
December			Completion of the presentation server Development of authoring tool Development of the student client tool
1999 January			
February	Report to the granting body	Testing and validation of the implementation	
March		Experiments in the use of the system	Development of the monitoring sub-system

T&L MET funding ceased

April	Publication on the commercial application of the developed technology	Employment of the monitoring sub-system	*The contracts for the research assistants expires*
June			Work to be continued by graduate students
August	Final reporting	Monitoring the tool use	
October	Preparation for a major grant in the area of the computer-aided education		
December		Project completion	

7. Related Work

Investigation of pattern-based approaches to problem solving dates back to 1964, when Christopher Alexander developed a collection of patterns in the field of Architecture and Urban Planning. Software design patterns are a relatively new approach to object-oriented software development (1987). A few books have been written about design, architectural and analysis patterns (Gamma et al, Bushmann et al, Fowler). Annual conferences on design patterns are held since 1993. Pattern design methods in Computer Science and Information Systems have been taught in several universities world-wide, e.g. at Carnegie Mellon University, Washington University, Tel Aviv University, California Polytechnic State University, and in the Department of Information Systems at the University of Melbourne. Organisational patterns have been studied and collected by Jim Coplien of Bell Labs and Martin Fowler of Citibank. Catalogues of software patterns are accessible over the web (e.g. http://st-www.cs.uiuc.edu/users/patterns/catalogs/). There have been no previous attempts to assist IS teaching with a comprehensive repository of problem-solving patterns. There are no comparable products, methods or technologies in this area.

8. Summary

This paper described the multimedia project that aims at more effective teaching of systems analysis and design to Information Systems students. The projects proposes to develop a multimedia repository of software analysis and design patterns explaining the main methods and techniques of solving typical software development problems. The proposed system will also facilitate a decision support system capable of guiding the students through the design process.

9. References

1. ACM (1995): *Proceedings of ACM Multimedia'95*. San Francisco, California: Addison-Wesley Pub Co.
2. Andris, J.F. (1996): *The relationship of indices of student navigational patterns in a hypermedia geology lab simulation to two measures of learning style.* Journal of Educational Multimedia and Hypermedia. **5**(3/4): p. 303-315.
3. Beattie, K., C. McNaught, and S. Wills (1994): *Interactive multimedia in university education: designing for change in teaching and learning: proceedings of the IFIP*. in *IFIP TC3 WG3.2 Working Conference on the Design, Implementation and Evaluation of Interactive Multimedia in University Settings*. Melbourne Victoria, Australia: Elsevier, p. .
4. Bosco, J. (1986): *An analysis of evaluations of interactive video*, in *Educational technology*. p. 7-17.
5. Brownstein, E.M.e.a. (1993): *Author Your Own!! Friendlier Software for Your Instructional Power*. in *The Annual Meeting of the National Science Teachers Association*. Kansas City, MO, p. .
6. Carlson, H.L. (1987): *Learning style and program design in interactive multimedia.* Educational Technology Research and Development. **39**(3): p. 41-48.
7. Carlson, H.L. and D.R. Falk (1990): *Effectiveness of interactive videodisc instructional programs in elementary teacher education.* Journal of Educational Technology Systems. **19**(2): p. 151-163.
8. Crowe, M.K. (1995): *Cooperative Work with Multimedia (Research Reports Esprit)*: Springer-Verlag.
9. Cybulski, J.L. (1996): *Introduction to software reuse*, TR 96/4, The University of Melbourne: Parkville.

10. Cybulski, J.L. (1996): *Reuse in the eye of its beholder: cognitive factors in software reuse*. in *OzCHI'96*. Hamilton, New Zealand: IEEE Press, p. 228-235.

11. Dodds, P.V.W. (1995): *The Digital Multimedia Cross Industry Guide*: Focal Pr.

12. Ebert, E. and N. Strudler (1996): *Improving Science Learning Using Low-Cost Multimedia.* Learning and Leading with Technology. **24**(1): p. 23-26.

13. Ellis, D. (1994): *Barefoot multimedia, or, all is not what it seems, moriarty*. in *IFIP TC3 WG3.2 Working Conference on the Design, Implementation and Evaluation of Interactive Multimedia in University Settings*. Melbourne Victoria, Australia: Elsevier, p. 151-154.

14. Fitzgerald, G.E., B. Wilson, and L.P. Sehrau (1997): *An interactive multimedia program to enhance teacher problem solving skills based on cognitive flexibility theory: Design and outcomes.* Journal of Educational Multimedia and Hypermedia. **6**(1): p. 47-76.

15. Foley, A.D. (1993): *Multimedia: Exploring Uses in Business Education.* Business Education Forum. **47**(3): p. 31-33.

16. Fowler, M. (1997): *Analysis Patterns: Reusable Object Models*. Menlo Park, California: Addison-Wesley.

17. Gamma, E., R. Helm, R. Johnson, and J. Vlissides (1995): *Design Patterns: Elements of Reusable Object-Oriented Software*. Readings, Massachusetts: Addison-Wesley.

18. Goforth, D. (1994): *Learner Control = Decision Making + Information: A Model and Meta-Analysis.* Journal of Educational Computing Research. **11**(1): p. 1-26.

19. Hannafin, R.D. and H.J. Sullivan (1995): *Learner Control in Full and Lean CAI Programs.* Educational Technology Research and Development. **43**(1): p. 19-30.

20. Holyoak, K.J. (1990): *Problem solving,* in *An Invitation to Cognitive Science: Thinking*, D.N. Osherson and E.E. Smith, Editors. The MIT Press: Cambridge, Massachusetts. p. 117-146.

21. Jacobson, M.J., C. Maour, P. Mishra, and C. Kolar (1995): *Learning with hypertext learning environments: Theory, Design and Research.* Journal of Educational Multimedia and Hypermedia. **4**(4): p. 321-364.

22. Jensen, M.S., K.J. Wilcox, J.T. Hatch, and C. Somdahl (1996): *A Computer-Assisted Instruction Unit on Diffusion and Osmosis with a Conceptual Change Design*, , .

23. Khalili, A. and L. Shashaani (1994): *The effectiveness of computer applications: A meta-analysis.* Journal of Research on Computing in Education. **27**(1): p. 48-61.

24. Kinzie, M.B.e.a. (1992): *Motivational and Achievement Effects of Learner Control over Content Review within CAI.* Journal of Educational Computing Research. **8**(1): p. 101-114.

25. Kozma, R.B. (1994): *A Reply: Media and Methods*.

26. Kulik, J.A., R.L. Bangert, and G.W. Williams (1983): *Effects of computer-based teaching on secondary school students.* Journal of Educational Psychology. **75**(1): p. 19-26.

27. Kulik, J.A., C.-L.C. Kulik, and P.A. Cohen (1980): *Effectiveness of computer-based college teaching: A meta-analysis of findings.* Review of Educational Research. **50**(4): p. 525-544.

28. Morris, S.J. and A.C.W. Finkelstein (1994): *Development of Multiple Media Documents*, , Imperial College, Department of Computing.

29. Najjar, L.J. (1996): *Multimedia information and learning.* Journal of Educational Multimedia and Hypermedia. **5**(2): p. 129-150.

30. Packard, A.L.e.a. (1995): *Evolution of a Instructional Design*. in *The Annual Convention of the American Educational Research Association*. San Francisco, CA, p. .

31. Rachal, J.R. (1993): *Computer-Assisted Instruction in Adult Basic and Secondary Education: A Review of the Experimental Literature, 1984-1992.* Adult Education Quarterly. **43**(3): p. 165-172.

32. Romiszowski, A.J. (1994): *Educational Systems Design Implications of Electronic Publishing.* Educational Technology. **34**(7): p. 6-12.

33. Saidi, H. (1994): *The Impact of Advance Organizers upon Students' Achievement in Computer-Assisted Video Instruction.* Journal of Educational Technology Systems. **22**(1): p. 29-38.

34. Schank, R.C. (1994): *Active learning through multimedia*, in *IEEE Multimedia*. p. 69-78.

35. Schramm, R.M. and G.E. Rich (1993): *CAI vs. Textbook for Grammar and Punctuation Skills.* Business Education Forum. **47**(4): p. 48-52.

36. Scott, C.J. (1994): *Applied Adult Learning Theory: Broadening Traditional CME Programs with Self-Guided, Computer-Assisted Learning.* Journal of Continuing Education in the Health Professions. **14**(2): p. 91-99.

37. Shapiro, J. (1996): *Collaborative Computing: Multimedia Across the Network*: Ap Professional.

38. Shin, E.C.e.a. (1994): *Effects of Learner Control, Advisement, and Prior Knowledge on Young Students' Learning in a Hypertext Environment.* Educational Technology, Research and Development. **42**(1): p. 33-46.

39. Swan, K. (1997): *Exploring the role of video in enhancing learning from hypermedia.* Journal of Technology Systems. **25**(2): p. 179-188.

40. Toomey, R. and K. Ketterer (1995): *Using Multimedia as a Cognitive Tool.* Journal of Research on Computing in Education. **27**(4): p. 472-82.

41. Wales, C. (1995): *Paving In-Roads for Technology Integration--a Classroom Example.* Learning and Leading with Technology. **23**(2): p. 41-42.

42. Wang, P. and P.-S. Chan (1995): *Advantages, Disadvantages, Facilitators, and Inhibitors of Computer-Aided Instruction in Singapore's Secondary Schools.* Computers and Education. **25**(3): p. 151-162.

43. Woolf, B.P. and W. Hall (1995): *Multimedia pedagogues: Interactive systems for teaching and learning.* Computer. **29**(5): p. 74-80.

44. Yang, Y.-C. and W.-K. Chin (1997): *Motivational analysis on the effects of type of instructional control on learning from computer-based instruction.* Journal of Technology Systems. **25**(1): p. 25-35.