# Considering a Full Range of Teaching Techniques for Use in Interactive Educational Software: A Practical Guide and Brainstorming Session

*Anne Morgan Spalter, Rosemary Michelle Simpson, Michael Legrand, and Saori Taichi*[1]

*Abstract* — *This paper addresses a moment in the design of educational software that is crucial and yet often unrecognized and unexploited: the moment when the designers decide what traditional teaching techniques (and related pedagogies) will be used. In our experience, this decision is often reached without due consideration of the alternatives, with the result that many educational modules tend to approach their subject matter in the same few ways, e.g., an interactive visualization of an algorithm or physical process. Our goal is not to recommend one teaching approach over another, but to reinforce awareness of the breadth of choice that is available. To that end, we have selected thirteen common teaching techniques that are particularly appropriate for interactive software and describe them in a traditional context and in terms of their potential online incarnations.*

*Index Terms* — *Teaching techniques, pedagogy, educational software, interface design.*

## INTRODUCTION

This paper addresses a moment in the design of educational software that is crucial and yet often unrecognized and unexploited: the moment when the designers decide what traditional teaching techniques (and related pedagogies) will be used. In our experience, this decision is often reached without due consideration of the alternatives. The result, especially in math, engineering, and science education software, is that many educational modules tend to approach their subject matter in the same few ways, e.g., an interactive visualization of an algorithm or physical process.

Our goal is not to recommend one teaching approach over another, but to reinforce awareness of the breadth of choice that is available. We hope that the experience of reading this paper will be similar to attending a brainstorming session-that readers will finish excited about the possibilities and ready to experiment more freely in their own work. If this paper is a success, it will be because designers of educational software, whether in K-12, universities or industry, feel inspired to pull out these descriptions and quickly read through them before designing their products.

The right solution may not be one single approach at all. Research into multiple learning styles [10] confirms what many teachers have already observed informally—that different students learn differently. The teaching techniques we discuss here provide for a range of learning styles, focusing on aspects from spatial relations to numerical-logical reasoning to interpersonal and social explanations.

We have selected thirteen common teaching techniques [13] that are particularly appropriate for interactive software and describe them in a traditional context and in terms of their potential online incarnations. A demonstration educational module illustrates each technique, with variations tuned to the specific approach. (In practice, few educational modules will use only one technique, most use two or three.) Our demonstration educational software module teaches basic concepts related to the acceleration of an object, such as a ball, in a force field, such as gravity.

The teaching techniques, arranged in an order that we feel best presents the range of possibilities for our demonstration module, are: Laboratory, Visualization, Simulation, Lecture and Demonstration, Case-Study, Role-Playing, Mastery Learning, Creative Project, Student Teaching, Playground, Drill, Behavior Modification, and Incidental Learning.

A note about the illustrations: the choice of teaching technique and design of the user interface greatly influence each other. We provide example interface sketches for several of our descriptions in their original, hand-drawn form, in keeping with the brainstorming motif of the paper.

## RELATED AND PREVIOUS WORK

It is impossible to list all, or even close to all, related work because the scope of this paper spans the entire domain of educational technology. Some of the most directly related work is mentioned in the description of each technique, with bibliographic references in the Reference section at the end. For a more comprehensive listing, see the Educational Technology Timeline [8], which lists key work over the last forty years and provides bibliographic and URL references.

The descriptions and example in this paper extend ongoing work in the Exploratories Project at Brown University [8]. Exploratories are combinations of science museum-like environments and laboratories, realized as two- and three- dimensional explorable worlds. The Exploratories Project has a dual mission of creating

[1] Brown University/ Dpt. Of Computer Science, Box 1910/ Providence, RI 02912. Please send correspondence to ams@cs.brown.edu.
We would like to acknowledge the Exploratories project's generous sponsorship by the NSF Science and Technology Center for Computer Graphics and Scientific Visualization. We would also like to thank Adobe Systems, Inc. Advanced Network & Services, Alias|Wavefront, IBM, Intel, Microsoft, and Taco, Inc.

innovative educational content for the Web and documenting the experiences of creating such software in a Design Strategy Handbook. The Handbook is under development and elements of it can be found in papers discussing fine-grained educational module size [11], reusable hypertext structures in which to embed educational applets [21], and methods of integrating education software modules into established curricula [22].

The handbook draws on our own and others' experiences and includes guides, templates, and patterns, as in Gamma et al's object-oriented "Design Patterns" [9]. (Patterns are formalized methods for sharing expert strategies.) The examples here do not qualify as patterns but begin to supply some of the needed material. Topics in the Handbook include assessment of resources, descriptions of different pedagogical categories, patterns for implementation, indexed examples, and information on interface and interaction design.

## THE TEACHING TECHNIQUES

### Laboratory

***Definition***: A Laboratory provides a structured environment in which students undertake observations, experiments, and evaluations. It usually involves attempts to apply theories to real world situations.

For example, in a chemistry lab students mix chemicals and observe results. This process may be confused with Case-Study because of the problem-solving skills involved. However, the distinctive characteristic of laboratories lies in the experimentation. Instead of gathering large amounts of information to analyze it, as in a Case-Study, students come up with an experiment and then execute it. It is through gathering and analyzing data generated by the experiment that the students learn whether their hypotheses holds, not in the extensive gathering and analyzing of disparate data. A nice set of laboratory experiments can be seen at the University of Oregon Virtual Java Lab [29].

***Example***: As in the cannon applet in the aforementioned University of Oregon site, students shoot canon balls out of a canon, trying to hit a mean creature on the ground (see Figure 1). The setup involves an expectation and an experimental set of parameters, then the execution of the experiment. The advantages of experimenting virtually can be leveraged by letting students perform experiments that would be impossible or impractical to conduct in real life.

### Visualization

***Definition***: Software animates the behavior of a process, algorithm, equation or other phenomenon, and sometimes provides the opportunity to manipulate the parameters.

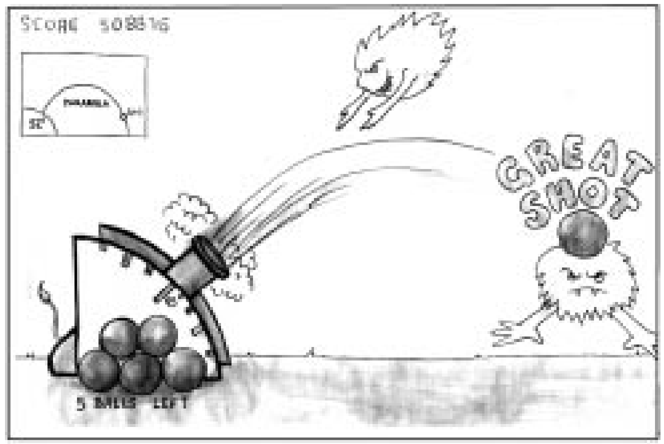


FIGURE. 1
LABORATORY EXAMPLE

Some visualizations may be simulations of real-life visual events and others may be purely for the sake of better understanding a concept. For example, it is helpful to visualize a sorting algorithm even if it the algorithm is never seen, visually, in the course of running a given program. Visualization of abstract concepts, such as color spaces, can help users better understand their forms and examine relationships within the spaces.

Visualization is frequently a component of math, engineering, and science educational software. Many concepts in these fields obviously benefit from visualization and the opportunity to make rapid changes and iterations and see immediate results. Tools such as Wolfram Research Mathematica and Key Curriculum Press Geometer's Sketchpad fall into this category, as do Professor Thomas Banchoff's four-dimension topological studies [2] and the algorithm animation work of Brown [6], Stasko [24], and Bridgeman [4].

***Example***: The equations governing force and acceleration are graphed. The user can type in new variables and see the different parabolas as an object's speed and angle of fire are changed. Alternatively, users can interact directly with the curves and then view the changed equations.

### Simulation

***Definition***: Simulation usually involves simulating an entire system, with properties and constraints selected to illustrate the principles being taught.

For example, wind tunnels are used to simulate atmospheric flight conditions for an object such as a space shuttle or an airplane wing model. On the computer, Simulation refers to the graphical and behavioral simulation of a physical system or environment. Only the relevant aspects of the real world need to be incorporated into the simulation, but there must be enough fidelity so that experiments and experiences in the simulation can be meaningfully related to real-world situations. Alan

Borning's ThingLab, which taught the physics of springs, was an early example of computer-based simulation [3].

***Example***: Unlike the oversimplified cannon applet (see Laboratory), a simulation module could have more a more realistic scenario with wind speeds, friction, and kickback from the cannon. One could use the Laboratory applet to understand the basic concept, but the Simulation would allow the user to do the calculations that would be used for a real cannon. Flight simulators are a popular example; many are based on real pilot training software.

### Lecture and Demonstration

***Definition***: Students passively acquire information that is presented by an expert.

When this information is purely verbal, the technique is Lecture, when visual, Demonstration. For example, a history teacher describing the events leading to World War II is giving a Lecture, while a biology teacher dissecting a cow's eye to show its different parts and functions is giving a Demonstration. In both cases, the distinguishing characteristic is that whatever is being told or shown is being done systematically and with great skill, and is watched passively by students.

The Lecture format can be pursued online with systems such as Gregory Abowd's Classroom 2000 [1], and variations which are found in many distance learning continuing education courses [23], [28]. This approach is used in our introductory programming course at Brown University [5] which places all its lecture notes and synchronized video or audio of the lectures online. (Note, the lectures are only available during the school year.) Although students can stop and start the lectures and navigate within them, they are still in a traditional lecture format

***Example***: A well thought out and entertaining lecture on gravity or electromagnetic theory by a top physicist is video-taped and put online. His lecture slides are also online and students can navigate to any part of the lecture and see the video and the slides at the same time (see Figure 2). The video could include experiments and Q&A from the audience. One could add interactive experiments that the student could do online (see Laboratory and Creative Project).

### Case Study

***Definition***: Students explore a set of materials representing a specific problem or situation, and use the analyze-generalize-synthesize problem-solving model to develop their own solutions.

For example, a teacher could pose the following question to her class: "Should Bell Atlantic make cell phones that can be installed inside people's heads, right under the scalp?" She would then allow the students to analyze the problem, break it down into all possible answers, collect data, and then synthesize the general information gathered to arrive at a conclusion. The goal is not for students to arrive at a predetermined right answer, but to learn how to analyze the given type of situation. This technique is used extensively in business schools and advanced history courses.

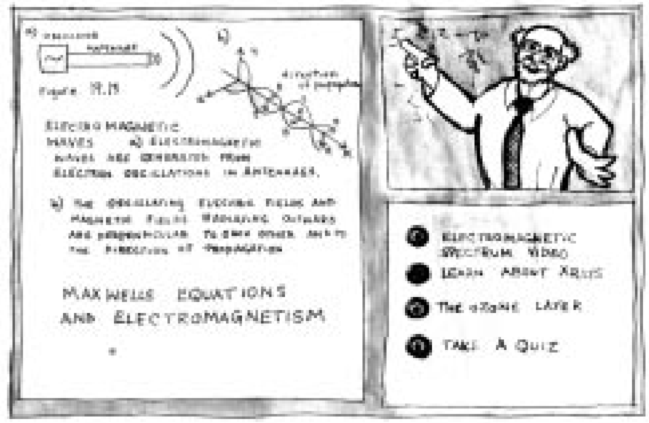


FIGURE. 2
LECTURE AND DEMONSTRATION EXAMPLE

An AI engine could be very useful for making this approach work in software when a teacher or facilitator is not intimately involved in the process. In this sense it is comparable to the role of cognitive tutors in such successful projects as CMU's PUMP (Pittsburgh Urban Mathematics Project), which is based on the ACT (Advanced Computer Tutoring) cognitive tutoring research [17].

***Example***: A low-budget rocket company wants to send its low- budget rockets into space. This module will help the user, who acts as company CEO, to analyze just how inexpensive the materials can be while still enabling the rocket to break out of Earth's gravitational field. The applet provides different vendors, with different materials, and a break-out-the-Earth's-gravitational-field calculator which must be used to ensure the viability of the purchasing decisions. In the process, the physics of different materials (weight, strength) are examined.

### Role Playing

***Definition***: To role play is to act under a new identity. The new identities are usually human roles but need not be.

For example, in ExploreNet [12] school children represent people from different cultures, while the activity of "playing turtle" in Logo provides an example of the use of role-playing in science and mathematics [18]. It is crucial that in using this technique students detach themselves from their realities and actively perform the role of another person, thing, or idea. This can be difficult to do on the computer effectively without a collaborative setup or some sort of AI, otherwise the process may lack richness. Historical roles could be part of an adventure game, however, and roles such as mayor of a city are used effectively in the game SIM City, which offers enough complexity to be compelling. Role Playing is used in training tools such as Roger Shank's conversion of a previously on-site business training course into an interactive computer-based experience [19].

***Example***: The user is Copernicus and must convince the court and church that the Earth revolves around the Sun, and not the other way around (see Figure 3). Use your physics knowledge and political savvy to change history. Such an approach would show students that physics facts are not achieved and applied in isolation from politics, economics, and religion.

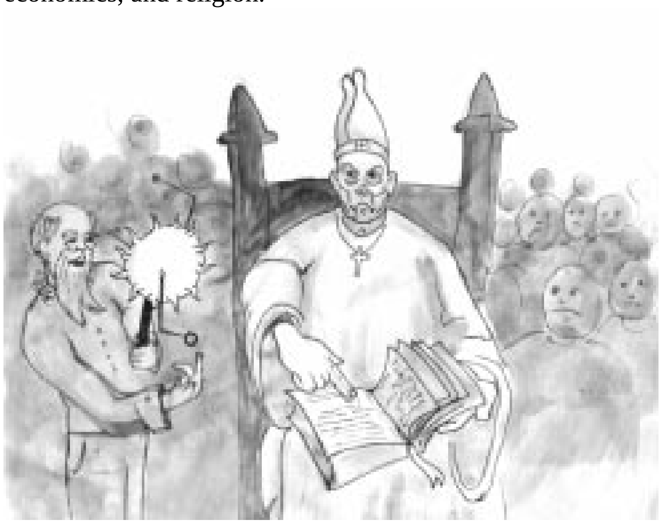

FIGURE. 3
ROLE PLAYING EXAMPLE

### Mastery Learning

***Definition***: A student advances to the next step only after mastering previous ones.

Teaching geometric proofs by presenting a new theorem only after a more basic one has been mastered is a good example [13]. In other words, there are prerequisites for everything that is taught. Mastery Learning is commonly employed in tutorials, which are often easy to do on a computer and which can include interesting interactive sections as well as multiple-choice questions. Some training development environments, such as Macromedia Authorware are designed to facilitate mastery learning.

***Example***: A tutorial provides text and video lectures on the concept of force, then acceleration, then gravity, and students are posed progressively more involved experiments to conduct or problems to solve. Although leaping ahead is not prohibited, the latter sections are, for all practical purposes, impossible to do without the knowledge gained in earlier sections.

### Creative Project

***Definition***: Creative Projects focus student activities around their creative capabilities such as painting, composing music, or designing gardens. Project outcomes can include reports or studies triggered by a student's personal interest.

Unlike more structured techniques, such as Drill or Mastery Learning, there is no right answer to "get", although there may be a goal, such as "make a painting that expresses your fear of the dark." There is a nice example of a creative project in the 1999 Advanced Network and Systems ThinkQuest competition in which users make movies, choosing, from a set of possibilities, their own shots and even one of several different endings [27].

***Example***: Users are offered the following challenge: given some equipment, which includes a fish tank, deck of cards, a toy sailboat, bellows, a 10lb weight, an arrow, some kerosine, and a wedge of Swiss cheese, build a Rube-Goldberg-type machine that will kill a mouse (see Figure 4). Students would have to calculate trajectories of the objects (students could make and test numerical calculations as well as interactively run their contraption and make adjustments).

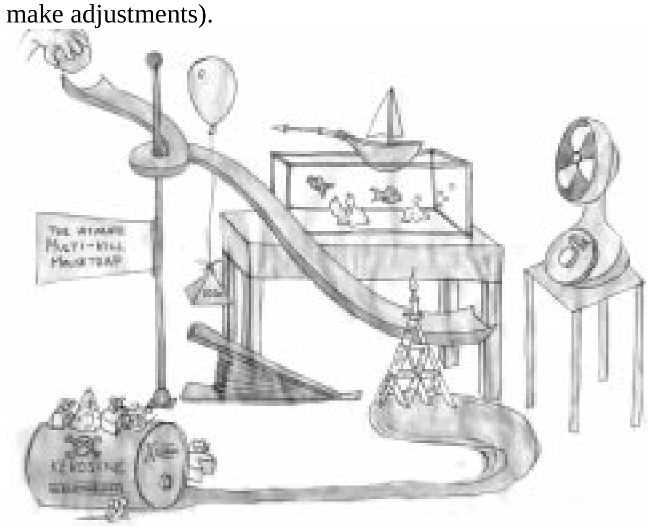


FIGURE. 4
CREATIVE PROJECT EXAMPLE

### Student Teaching

***Definition***: Students teach students, in a variety of situations.

For example, in Eric Mazur's Peer Instruction techniques, lectures are broken up into miniature concept tests in which pairs of students attempt to convince each other about the solution to a physics concept question [15].

In other scenarios, students read different portions of a book and summarize their sections for each other. This is also used when teaching by reflection, i.e., sometimes it is educationally valuable for a student to look back at a finished project and consider ways in which he could have done it better [19]. This creates an instance of a student teaching himself.

Student teaching takes place across the globe in the enormous ThinkQuest project [27]. The advantage may be that students present material in a way more conducive to other students than when adults structure the material. An obvious problem is lack of expertise, so it is usually essential to have an adult subject-matter expert involved.

***Example***: Students must create Web pages to teach each other about acceleration. They can use graphics, links to other sites, and interaction (as their programming ability allows). Templates can be provided requiring students to write definitions of terms, create illustrations, and choose links. There's nothing like explaining something to others to find out whether or not you understand it yourself! This requires some oversight, though, and involvement of a live teacher.

### Playground

***Definition***: A Playground provides a system or environment in which to explore and perform self-generated experiments.

It differs from Simulation in that the constraints are looser and do not necessarily bear any relation to 'real-world' laws, forces, and restrictions. As with science fiction and fantasy, however, there needs to be internal consistency in order to preserve predictability—"Beam me up, Scotty" should always result in the same consequence.

As in an actual playground, "toys" and a context in which to "play" with them are provided but guidance is usually minimal or nonexistent. There is no correct answer to reach or time limit for use. In additional to being educational, a Playground experience should be aesthetically pleasing and fun [20].

***Example***: A user enters a 3D world in which there are a number of buildings of various heights. The user employs a special calculator to change his or her effective size and weight. Then the user "jumps" off of and between buildings (see Figure 5). Changing one's weight and size affect how well one is able to jump to the desired locations. There are interesting things inside the buildings which help teach related topics.

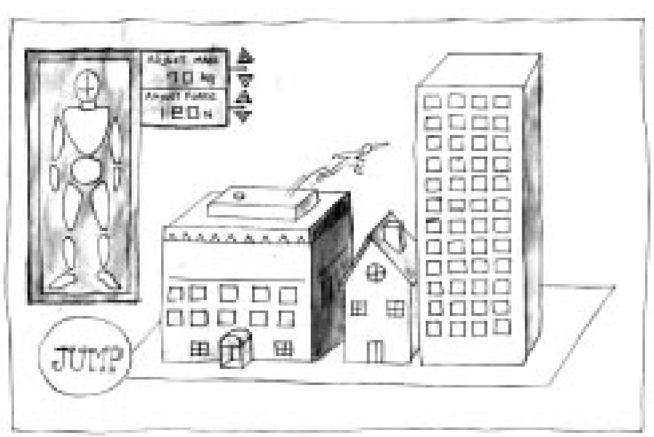


FIGURE. 5
PLAYGROUND EXAMPLE

### Drill

***Definition***: Multiple fine-grained and closely related exercises develop accuracy and speed of performance.

A simple example is drilling on the multiplication table. The chief goal of drilling is to internalize a certain skill or knowledge to the point where it becomes implicit. The computer can help make this as enjoyable as possible with interesting graphics and sounds and even a game-like structure. Electronic examples of Drill abound, most today also incorporating Incidental Learning or Behavior Modification by placing the drills in a game-like format or simply offering rewards for right answers. Some of the earliest work in computer-aided education, Patrick Suppes' CAI work at Stanford [26], employed this technique, and it is used today by many of the inexpensive commercially available math software packages [7].

***Example***: Students choose from multiple options to identify key equations about acceleration and gravity. They could also do a large number of simple calculations, using such equations, get instant feedback about their approach and final answers.

### Behavior Modification

***Definition***: After an objective is set, actions favorable to that objective are positively reinforced while detrimental actions are penalized.

For example, a correct approach to a problem could result in an entertaining noise or a exploding graphics and an incorrect approach could make the student repeat sections or accumulate fewer points in a game.

Behavior Modification is not just praise for right answers and criticism for wrong ones, though (see Drill), it must help students build intuition for a good approach to solving a problem or completing a task. For example, imagine a program which rewarded students for trying to solve an equation by moving terms of the same variable to one side while penalizing them for guessing randomly or moving terms in a way that doesn't make sense.

***Example***: In the example given in the Laboratory module (see Laboratory), students who repeatedly aim the canon in such a way that the ball misses the target might run out of balls to shoot with, to discourage random cannon angles. If the module also has a game-like approach (see Incidental Learning), the shooter might be overtaken by monsters. Students who systematically choose good settings and hit the target with only a few tries could be rewarded with flashy graphics or harder challenges with higher skill levels and recorded high scores.

### Incidental Learning

***Definition***: Students learn material in passing while achieving a fun and engaging goal.

Many subject areas are not reliably compelling for students; incidental learning lets the student learn such subjects in the course of doing some more, to them, desirable task. A real life example is to ask students to bake eight cookies by giving them a recipe for twelve. The task of baking cookies provides its own motivational forces, so as students measure the flour and sugar, they won't notice that they are incidentally learning about fractions and common denominators.

This approach is used in Droid Works by Lucas Learning [14]. In this game students build robots to save the galaxy from the Evil Empire and along the way learn about the principles behind energy, force, motion, simple machines, light and magnetism. Incidental learning could also take place in a Creative Project or freeform Playground (but is not usually guaranteed in the later). Roger Shank discusses some of the motivations and pitfalls of this technique in "Engines for Education" [19].

***Example***: You are trapped in the evil fortress Xulu and must escape with only a set of weighted metal balls, a length of yarn, a stick of chewing gum, and a vacuum cleaner (see Figure 6). Students must use the concepts of acceleration and gravity, but are kept interested by the game's goals, interesting graphics and sounds.

FIGURE. 6
INCIDENTAL LEARNING EXAMPLE

## ACKNOWLEDGMENT

We would like to thank Michael Legrand for his illustrations.

## REFERENCES

[1] http://vr-atlantis.com/lbe_guide/lbe_list3.html

[2] Beall J.E., Doppelt A.M., and Hughes, J.F., Developing an Interactive Illustration: Using Java and the Web to Make It Worthwhile. Proceedings of 3D and Multimedia on the Internet, WWW and Networks, 16-18 April 1996, Pictureville, National Museum of Photography, Film & Television, Bradford, UK, 1996.

[3] Bell, John T. and Fogler, H. Scott. "The Investigation and Application of Virtual Reality as an Educational Tool" in Proceedings of the American Society for Engineering Education,1995.
(http://www.engin.umich.edu/research/vrichel/)

[4] Benford, S., Taylor, I., Brailsford, D., Koleva, B., Craven, M., Fraser, M., Reynard, G., and Greenhalgh, C. "Three-dimensional Visualisation of the World Wide Web" in ACM Computing Surveys Symposium on Hypertext (2000).

[5] Bernstein, Mark. Storyspace: Hypertext Writing Environment. Eastgate Systems Press, Watertown, Massachusetts.

[6] Chen, Chaomei and Czerwinski, Mary. "From Latent Semantics to Spatial Hypertext - An Integrated Approach" in Proceedings of Hypertext '98, ACM SigLink, 1998.

[7] Fosnot, Catherine, ed. Constructivism: Theory, Perspectives, and Practice. New York: Teachers College Press, 1996.

[8] Gould, Daniel L., Simpson, Rosemary M., and van Dam, Andries. "Granularity in the Design of Interactive Illustrations'' in Proceedings of SigCSE '99, ACM SigCSE, 1999.

[9] Lindeman, Robert W., Sibert, John L., and Hahn, James K. "Towards Usable VR: An Empirical Study of User Interfaces for Immersive Virtual Environments" in Proceedings of SIGCHI '99, ACM SIGCHI, 1999.

[10] Loftin, R. Bowen and Kenney, Patrick J. "Virtual Environments in Training: NASA's Hubble Space Telescope Mission" in Interservice/Industry Training Systems & Education Conference, 1994.
(http://www.vetl.uh.edu/Aerospaceov.html;
http://www.vetl.uh.edu/Hubble/longpaper.html)

[11] Maes, Pattie. "Artificial Life Meets Entertainment: Lifelike Autonomous Agents" in Communication of the Association for Computing Machinery, V38, No. 11, November 1995.

[12] Moher, T., Johnson, A., Ohlsson, S., Gillingham, M., Bridging Strategies for VR-Based Learning In the proceedings of CHI 99, Pittsburgh, Pennsylvania, May 15-20, 1999, pp. 536-543, 1999.
(http://www.evl.uic.edu/roundearth/)

[13] Pierce, Jeffrey, Forsberg, Andrew, Conway, Matthew, Hong, Seung, and Zeleznik, Robert. Image Plane Interaction Techniques in 3D Immersive Environments. 1997 Symposium on Interactive 3D Graphics, pages 39-43.

[14] Pierce, Jeffrey, Stearns, Brian, and Pausch, Randy . Two Handed Manipulation of Voodoo Dolls in Virtual Environments. 1999 Symposium on Interactive 3D Graphics, pages 141-145.

[15] Rose, Howard and Billinghurst, Mark. Zengo Sayu: An Immersive Educational Environment for Learning Japanese. University of Washington, Human Interface Technology Laboratory, Report No. r-95-4, 1995.

[16] Salzman, J., Dede, C, and Loftin, B. "Learner-Centered Design of Sonorily Immersive Microworlds Using a Virtual Reality Interface" in Proceedings of the Seventh International Conference on Artificial Intelligence and Education, pp. 554-564, 1995.
(vetl.uh.edu/ScienceSpace/learnvir.html;
http://www.vetl.uh.edu/ScienceSpace/absvir.html)

[17] Simpson, Rosemary M., Spalter, Anne M., and van Dam, Andries. Exploratories: An Educational Strategy for the 21st Century", in Electronic Schoolhouse of ACM SIGGRAPH '99, 1999.

[18] Spalter, Anne Morgan and Simpson, Rosemary Michelle. "Reusable Hypertext Structures for Distance and JIT Learning" accepted for ACM Hypertext 2000, 2000.

[19] Spalter, Anne Morgan and Simpson, Rosemary Michelle. "Integrating Interactive Computer-Based Learning Experiences Into Established Curricula" submitted to ITICSE 2000, 2000.

[20] Winn, William. A Conceptual Basis for Educational Applications of Virtual Reality. University of Washington, Human Interface Technology Laboratory, Report No. TR-93-9, 1993.

[21] Zyda, M. and Sheehan, J. (eds). Modeling and Simulation: Linking Entertainment and Defense. National Research Council, National Academy Press, 1997.