DOCUMENT RESUME

ED 388 243 IR 017 392

AUTHOR Feifer, R.; Allender, L.
TITLE It's Not How Multi the Media, It's How the Media Is Used.
PUB DATE 94
NOTE 7p.; In: Educational Multimedia and Hypermedia, 1994. Proceedings of ED-MEDIA 94--World Conference on Educational Multimedia and Hypermedia (Vancouver, British Columbia, Canada, June 25-30, 1994); see IR 017 359.
PUB TYPE Reports - Descriptive (141) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Computer Simulation; Computer Software; Courseware; *Experiential Learning; *Multimedia Instruction; *Simulated Environment
IDENTIFIERS *Case Method (Teaching Technique)

ABSTRACT

Multimedia educational software is often a glitzy version of old technology. Some educational software has become better as developers began to ask, "In what ways can the computer facilitate learning, that were not possible before?" One answer to this question is: provide a simulated environment for the learner to interact with. For multimedia to have an impact on learning, a similar question must be asked: "In what ways can multimedia facilitate learning, that were not possible before?" One answer is the Case-Based Learn-by-doing Environments (CaBLE). The computer provides a simulated environment that allows the student to learn a task by doing a task. Multimedia stories and information help connect what the learner is doing in the simulated world with events in the real world. The CaBLE tutor utilizes the following components: task simulator; interface; library of failures which can occur in the domain; library of learner mistakes; learner state map; network of declarative knowledge; and library of stories. (Contains 21 references.) (Author)

It's Not How Multi The Media, It's How The Media Is Used

R. FEIFER
United States International University, 10455 Pomerado Road, San Diego, CA 92131
E-Mail: feifer@ils.nwu.edu

L. ALLENDER
Institute for the Learning Sciences
Northwestern University, 1890 Maple Avenue, Evanston, IL 60202, USA
E-Mail: allender@ils.nwu.edu

ABSTRACT: Multimedia educational software is often a glitzy version of old technology. This is not a new phenomena, early educational software fell into the same trap. Developers depended on the sexiness of the computer to make page-turning software more compelling. Some educational software has become better as developers began to ask, "In what ways can the computer facilitate learning, that were not possible before?" One answer to this question is: provide a simulated environment for the learner to interact with.

For multimedia to have an impact on learning we must ask a similar question, "In what ways can multimedia facilitate learning, that were not possible before?" One answer is what we call Case-Based Learn-by-doing Environments (CaBLE). The computer provides a simulated environment that allows the student to learn a task by doing the task. Multimedia stories and information helps connect what the learner is doing in the simulated world with events in the real world.

THE DARK AGES OF EDUCATIONAL SOFTWARE

Traditional computer aided instruction often use the following paradigm: the learner reads some content and then is tested on his knowledge; if the learner fails the test, then the process is repeated.

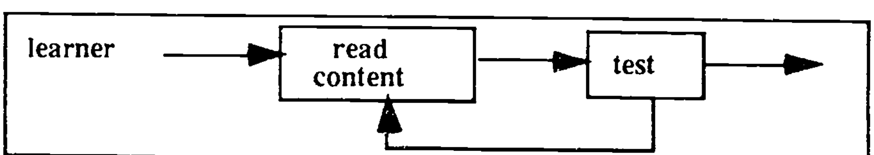

Figure 1. learner interaction with some instructional software

In this paradigm, the technology is used to evaluate the learner's test answers and to generate new lessons which may provide remedial information. Even given the best of implementations, this paradigm has the following problems:

1. the only motivation for reading the content is to pass the test;
2. the learner is exposed to the content often without an understanding of the context in which it would apply;
3. the approach emphasizes applying the right label to a concept, rather than the use of that concept in an appropriate situation.

A designer might attempt to augment this paradigm by delivering the initial content in a multimedia format, or presenting custom multimedia feedback for each wrong answer in the test. Although a multimedia version of the paradigm is more appealing, it will have the same basic flaws.

EXPLOITING COMPUTERS

Some designers have improved on this paradigm by utilizing the computer to create a simulated task environment for the student. Researchers have argued that the best way to learn how to do something is to try to do it and learn by your mistakes (Collins, Brown, & Newman, 1989; Laird, Rosenbloom, & Newell, 1986; Saxe, 1992; Schank & Jona, 1991; VanLehn, 1988). So instead of having learners simply read the content, learners take real life actions.

Simulated task environments broaden the types of training available. Actions in real life may be too expensive or dangerous for a student to attempt without the proper skills (i.e., flying a plane or disarming a bomb), but on a computer learners can actually learn by doing in these otherwise inaccessible task environments, (Govindaraj, 1988; Lesgold, Lajoie, Bunzo, & Eggan, 1991; Schauble, Glaser, Raghavan, & Reiner, 1991; Williams, Hollan, & Stevens, 1981).

Although simulations can make learning by doing an efficient instructional design, three factors limit the use of computer simulations in teaching:

1. good simulations are hard to build;
2. learners can flounder with just a simulation;
3. learners may not believe the simulation.

Building a good simulation in any but the most trivial domains is difficult. Whether you are simulating a physical system or a social system, allowing the user a wide range of actions and simulating the results of any combination of those actions requires a complete model of the domain.

Even if we could build a simulation with sufficient fidelity, it is difficult to learn from mistakes without some guidance (Kuhn, 1989; Schauble, et al., 1991) In particular, without guidance, learners who have flawed strategies may fail to successfully learn the skills targeted by the simulation. Without guidance, failure can only show you what not to do. It is often not obvious why you failed or what you should have done instead.

One solution to this floundering problem is the addition of a computer-based coach to watch over the learner's shoulder and advise (Burton & Brown, 1976; Goldstein, 1979). The coach, however, also requires a good model of the domain, plus a model of the learner. Even if we could model a learner's misunderstandings in a simulation, we still have the problem of generating a dialog to ameliorate those misunderstandings.

The last problem is that a student may not believe that a simulation is accurate. After all, even the best simulations are not a complete representation of reality. Multimedia technologies can link the simulation to real events.

EXPLOITING MULTIMEDIA TECHNOLOGY

We have used multimedia to provide this link to reality in a model we call Case-Based Learn-by-doing Environments (CaBLE). In a CaBLE tutor, instead of the computer generating text based instruction, good story tellers, experts in the domain, tell their stories on video tape. These stories are then indexed to the kinds of failure for which they are relevant. This enables the system to react to the learner's failures as a good teacher might, by recalling a real and personal story containing the principles to be learned from the failure.

These stories make teaching through simulations more practical in three ways:

1. the simulation need only provide a context and motivation for the story, the stories make up for any lack of depth or fidelity in the simulation;
2. it is easier to index failures than to model the learner sufficiently to provide intelligent coaching;
3. it is easier to show a video than to generate instruction, and more compelling to the learner;

The interaction with a CaBLE tutor is illustrated in Figure 2. The test from Figure 1 is replaced in a CaBLE tutor with the simulation of a real-world task. There is no test in the conventional sense of the word, successfully completing the task is evidence of mastery. In addition, the order from figure 1 is reversed. The task comes first. The learner is exposed to background material only later, whenever he/she wants or needs it.

If the learner does not know what to do, he/she can ask questions that are answered either by information declarative information or a story. If the learner attempts the task but experiences some failure, the tutor will present a story. The tutor chooses a story that will either help the learner explain the failure, or discover a strategy for avoiding the failure in the future, or both.

MAKING IT REAL

In a CaBLE tutor, multimedia technology strengthens the connection between what is being taught, and reality. If a learner makes a mistake in a conventional tutoring system, and the system merely tells him that he has made a mistake, it will have little impact. The learner may not even believe that it was truly a mistake. Maybe the computer did not understand what he did.

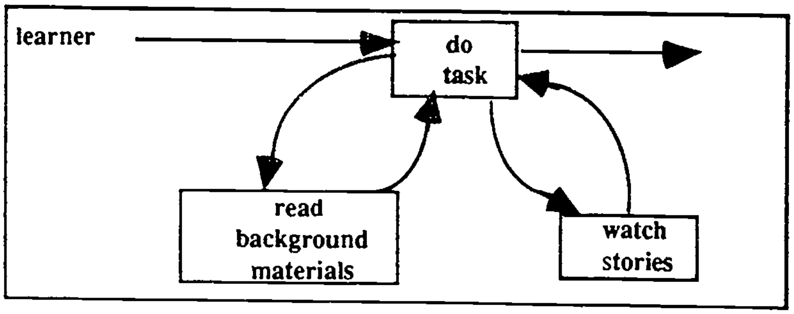

Figure 2. learner interaction with a CaBLE Tutor

If we wait for the mistake to lead to a failure within a simulated world, and then tell the learner that the mistake led to the failure, there is a better chance that the learner will see why it is a mistake. The learner will at least see why, in the tutor's opinion, it is a mistake. He may still, however, doubt that the linkage between the mistake and the failure would occur in the real world.

In a CaBLE tutor we go one step further. After the mistake has led to a failure in the simulated world, the system presents a video of a person describing a first person experience with a similar mistake, and the real-world failure that it led to. Although we could use text to tell the story, a video of a real expert provides the same information in a more credible format.

CABLE PARTS

A CaBLE tutor utilizes the following components: a task simulator, an interface, a library of failures which can occur in the domain, a library of learner mistakes, a learner state map, a network of declarative knowledge and a library of stories.

SIMULATION

The simulation of the task indicates the probable results of any learner action, if that action had occurred in the real world. The CaBLE tutor checks after each learner action to see if the mistake library recognizes that action as a mistake and to see if any resulting events are recognized by the failure library.

INTERFACE

The interface provides the learner with the means to take actions he/she might expect to be able to take in the actual task. In addition the interface communicates the results of the actions as determined by the simulation. The interface also contains generic question buttons that are interpreted by the current learner state to provide access to the knowledge network.

FAILURE LIBRARY

The failure library contains a list of events that would be considered failures in the domain (i.e., plane crashing or patient dying). Any time an event occurs in the simulation that matches one of these failure events the CaBLE tutor will check to see what mistakes have occurred that could have led to that failure.

MISTAKE LIBRARY

The mistake library contains a list of the common errors made in the domain. For each mistake there is a set of conditions that would allow the CaBLE tutor to recognize that the error occurred. In addition, each mistake is associated with a list of failures that could occur if the learner made the particular mistake. Each mistake is linked to appropriate stories and/or information in the knowledge network so that these can be offered to the learner after a failure.

LEARNER STATE MAP

The learner state map allows the CaBLE tutor to recognize the context in which the learner asks a question. This allows the tutor to provide more relevant information in response to the question. The learner's current location in the learner state map is updated when key interface actions occur and when key events occur in the simulation.

KNOWLEDGE NETWORK

The Knowledge Network contains the declarative knowledge the learner would have to have to be successful at the task. When the learner asks a question, the CaBLE tutor will bring them to an appropriate piece of information in the network.

STORY LIBRARY

The story library contains explanations, positive examples and negative examples for the domain being taught. When a failure occurs, the CaBLE tutor finds and tells a relevant story.

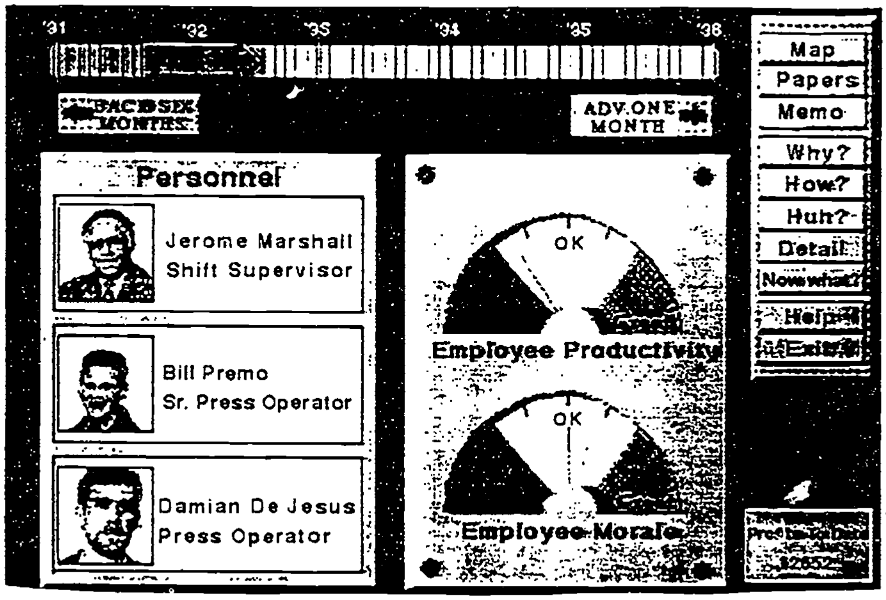

Figure 3: HeRMiT Opening Screen

AN EXAMPLE CABLE SYSTEM

HeRMiT is part of a business practices course designed to teach the principles of human resource management (HRM). The learner is asked to accomplish the following task (Bell & Feifer, 1992; Feifer & Hinrichs, 1992):

> You are the manager of the human resources department of the case company. You must make human resource decisions for the next four years, without destroying the company. To make your job easier, you will only make decisions for three employees. The computer will generalize your decisions to the rest of the staff.
>
> You can monitor the health of the company by watching the productivity and morale meters. If either of these meters goes into the red, you will be fired.

Behind this task is a simple simulation of employees and how their attitude, competence and performance interact to affect the overall success of a company. The learner must make appropriate long-term decisions for each of three employees, as well as respond to ad hoc situations as they arise (see figure 3).

The learner opens an employee's personnel file by clicking on their picture button on the main screen. The learner determines how the employee is doing by checking their static information, current attitude, competence or performance, paper trail, or history. They can also get more information as the result of initiating a formal evaluation or an informal "counseling" session. The learner takes action by changing the employee's position, salary, status or scheduling training. When the learner decides that they have taken all appropriate actions for a given month, they click on the "advance 1 month" button.

ASKING QUESTIONS

If the learner knows everything there is to know about HRM, he/she continues through the simulation rather quickly. If the learner lacks the basic principles, he/she can ask the tutor questions. The combination of the question button pressed and the learner's current context, permits the tutor to show an appropriate piece of information (Jona, Bell, & Birnbaum, 1991) If the information shown raises further questions, the learner can press a follow-up question button.

UNDERSTANDING FAILURE

When the learner makes mistakes he/she receives the same response he/she would receive in the real world: productivity goes down or employees become unhappy. When an employee quits or the company goes bankrupt, the tutor helps the learner understand what actions led to the failure and why. The learner is first shown a list of mistake types that are possible causes for the failure (Figure 4).

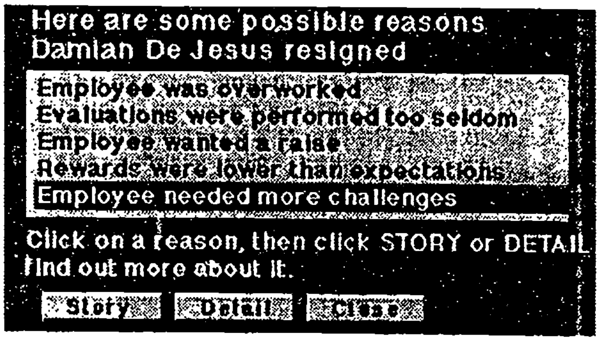

Figure 4. Failure Dialog Box

For each of these mistake types the learner can see:

- text information that explains why this type of mistake can be a cause of this failure;
- a video-taped story that provides a concrete example of this mistake type causing a similar failure.

SUMMARY

CaBLE tutors are designed to better exploit the potential of computers and multimedia technology.

The computer is exploited to provide a task that is challenging and interesting. We have found little need for any extrinsic motivation to induce a learner to sit down and attempt the task presented in HeRMiT. Further, once involved in the task, they will stay with it until they succeed.

But the above can also be said of most video games. In order to be successful in the task the student must know and/or learn whatever it is we want to teach. The tasks are designed so that it is impossible to take random actions and successfully achieve the goal. The task is the test.

Hypermedia allows the learner who cannot accomplish the goal with their existing knowledge, to learn new knowledge. They learn only on demand, and only in a context in which the learning will be meaningful.

When a learner takes an action that leads to a failure, multimedia allows the learner to see a real person telling a real example of that mistake and its consequences.

REFERENCES

Bell, B. L., & Feifer, R. G. (1992). Intelligent tutoring with dumb software. In C. Frasson, G. Gauthier, & G. I. McCalla (Eds.), *Intelligent Tutoring Systems* Berlin: Springer-Verlag.

Burton, R., & Brown, J. S. (1976). A tutoring and student modeling paradigm for gaming environments. ACM SIGCSE (Computer Science and Education) Bulletin, 8, 236-246.

Collins, A., Brown, J. S., & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L. B. Resnick (Eds.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 453-494). Hillsdale, NJ: Lawrence Erlbaum Associates.

Elliott, E. S., & Dweck, C. S. (1988). Goals: An approach to motivation and achievement. *Journal of Personality and Social Psychology*, 54(1), 5-12.

Feifer, R. G., & Hinrichs, T. R. (1992). Using stories to enhance and simplify computer simulations for teaching. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, . Indianapolis, Indiana: Lawrence Erlbaum Associates.

Feifer, R. G., & Soclof, M. S. (1991). Knowledge-based tutoring systems: Changing the focus from learner modelling to teaching. In *Proceedings of the International Conference on the Learning Sciences*, (pp. 151-157). Northwestern University, Evanston, IL: Association for the Advancement of Computing in Education.

Goldstein, I. P. (1979). The genetic graph: A representation for the evolution of procedural knowledge. *International Journal of Man-Machine Studies*, 11, 51-77.

Govindaraj, T. (1988). Intelligent computer aids for fault diagnosis training of expert operators of large dynamic systems. In J. Psotka, L. D. Massey', & S. A. Mutter (Eds.), *Intelligent tutoring systems: Lessons learned* (pp. 303-321). Hillsdale, NJ: Lawrence Erlbaum Associates Publishers.

Jona, M., Bell, B., & Birnbaum, L. (1991). Button theory: A taxonomy of student-teacher communication for interface design in computer-based learning environments. In *Proceedings of the Thirteenth Conference of the Cognitive Science Society*, . Chicago, IL.:

Kuhn, D. (1989). Children and adults as intuitive scientists. *Psychology Review*, 96(4), 674-689.

Laird, J. E., Rosenbloom, P. S., & Newell, A. (1986). *Universal subgoaling and chunking: The automatic generation and learning of goal hierarchies*. Hingham, MA: Kluwer Academic Publishers.

Lesgold, A., Lajoie, S., Bunzo, M., & Eggan, G. (1991). SHERLOCK: A coached practice environment for an electronics troubleshooting job. In J. H. Larkin & R. W. Chabay (Eds.), *Computer assisted instruction and intelligent tutoring systems: Shared goals and complementary approaches* (pp. 201 - 238). Hillsdale, NJ: Erlbaum.

Norman, D. A. (1986). Cognitive engineering. In D. A. Norman & S. W. Draper (Eds.), *User Centered System Design* (pp. 31-61). Hillsdale, NJ: Lawrence Erlbaum Associates.

Saxe, G. B. (1992). Studying children's learning in context: Problems and prospects. The *Journal of the Learning Sciences*, 2(2), 215-234.

Schank, R. C. (1991). Case-based teaching: Four experiences in educational software design (Technical Report No. 7). The Institute for the Learning Sciences, Northwestern University, Evanston IL.

Schank, R. C., & Jona, M. Y. (1991). Empowering the student: New perspectives on the design of teaching systems. The *Journal of the Learning Sciences*, 1(1), 7-35.

Schauble, L., Glaser, R., Raghavan, K., & Reiner, M. (1991). Causal models and experimentation strategies in scientific reasoning. The *Journal of the Learning Sciences*, 1(2), 201-238.

Seligman, M. E. P. (1975). *Helplessness: On depression, development, and death*. San Francisco: Freeman.

VanLehn, K. (1988). Toward a theory of impasse-driven learning. In H. Mandl & A. Lesgold (Eds.), *Learning issues for intelligent tutoring systems* (pp. 19-41). New York: Springer-Verlag.

Williams, M. D., Hollan, J. D., & Stevens, A. L. (1981). An overview of STEAMER: an advanced computer-assisted instruction system for propulsion engineering. *Behavior Research Methods and Instrumentation*, 13(2), 85-90.

Ziegler, R. (1992). Worldwide technology firm reaps returns on interactive multimedia training. In *Interactive Multimedia: return on investment analysis for learning and communication* (pp. 13-15). San Francisco, CA: Macromedia.

ACKNOWLEDGMENTS

This research was supported in part by the Defense Advanced Research Projects Agency, monitored by the Air Force Office of Scientific Research under contract F49620-88-C-0058 and the Office of Naval Research under contract N00014-90-J-4117, by the Office of Naval Research under contract N00014-J-1987, and by the Air Force Office of Scientific Research under contract AFOSR-89-0493. The Institute for the Learning Sciences was established in 1989 with the support of Andersen Consulting, part of The Arthur Andersen Worldwide Organization. The Institute receives additional support from Ameritech and North West Water, Institute Partners.

In addition to the authors, the following people were involved in the development of HeRMiT: Larry Langelier, Donna Fritzsche, Wayne Schneider, Michael Korcuska, Tom Hinrichs, Joshua Tsui, Carolyn Caballero, Leena Nanda, Cheryl Jindra, Matt Greising, Alan Nowakowski, and Don Jastrebsky. Special thanks to the HeRMiT storytellers: Bill Braemer and Tim Coan.