DOCUMENT RESUME

ED 383 317 IR 017 172

AUTHOR	Law, Michael P.; And Others
TITLE	Developing Electronic Performance Support Systems for Professionals.
PUB DATE	95
NOTE	8p.; In: Proceedings of the 1995 Annual National Convention of the Association for Educational Communications and Technology (AECT), (17th, Anaheim, CA, 1995); see IR 017 139. Figure may not reproduce well.
PUB TYPE	Reports - Evaluative/Feasibility (142) -- Speeches/Conference Papers (150)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	Case Studies; *Computer Assisted Testing; Educational Assessment; *Evaluation Methods; Teacher Education
IDENTIFIERS	*Alternative Assessment; *Performance Support Systems

ABSTRACT

This paper discusses a variety of development strategies and issues involved in the development of electronic performance support systems (EPSS) for professionals. The topics of front-end analysis, development, and evaluation are explored in the context of a case study involving the development of an EPSS to support teachers in the use of alternative assessments. Strategies and concepts such as rapid prototyping, formative experimentation, usability, and socio-technical perspectives are highlighted. EPSS developers utilize principles found in instructional systems, software engineering, performance technology, and formative experimentation to develop an effective system for teachers. By making usability a goal and focusing on the social, organizational, and cultural factors that influence how work is performed by professionals in a specific work context, developers increase the likelihood of developing effective EPSSs that support the people and organization in meeting their goals. A figure illustrates the system prototype. (Contains 20 references.) (Author/MAS)

Title:

Developing Electronic Performance Support Systems for Professionals

Authors:

Michael P. Law, James R. Okey, and Bryan J. Carter
Department of Instructional Technology
The University of Georgia
607 Aderhold Hall
Athens, GA 30605

Abstract

This paper discusses a variety of development strategies and issues involved in the development of electronic performance support systems for professionals. The topics of front-end analysis, development, and evaluation are explored in the context of a case study involving the development of an EPSS to support teachers in the use of alternative assessments. Strategies and concepts such as rapid prototyping, formative experimentation, usability, and socio-technical perpectives are highlighted.

Developing Electronic Performance Support Systems for Professionals

Professionals come to their jobs in today's rapidly changing workplace from a variety of different backgrounds each with a collection of different interests, skills, and expertise. As a result, educational planners see the need to provide multiple and alternative educational experiences, rather than try to design the one best training program (Scribner & Sachs, 1991). A method for meeting this challenge is to provide people learning opportunities and experiences on-the-job and at the time of need to bridge knowledge gaps and skill deficiencies encountered while performing their jobs (Geber, 1990; Gery, 1991; Law, 1994).

A new tool integral to the success of this approach is what has been coined by Gery (1991) and others as an Electronic Performance Support System (EPSS). In its most basic form, an EPSS is a computer-based performance tool consisting of an informational database, an advisory system, models and examples, and a tutorial/learning base composed of granular-sized pieces of computer-based training all linked to one another in a relational database. The primary characteristic of this system that allows the user to solve problems due to a lack of knowledge or skills is the ability to quickly access well-structured pieces of information, advice, models, and tutorial assistance at the moment of need as determined by the user in the context of the present performance problem. These systems range in complexity and sophistication based on the development resources available and the nature of the performance support context. To date, most of these systems have been developed for use in business, industrial, and governmental and military settings, however, this appears to be changing. Instructional technologists and other educators are examing ways in which these systems can be developed to support teaching and learning in educational settings (Brush, Knapczyk & Hubbard, 1993; Collis & Verwijs, 1995; Goodrum, Dorsey & Schwen, 1993; Schwen, Goodrum & Dorsey, 1993).

The Learning and Performance Support Laboratory at the University of Georgia is currently developing an EPSS to support school teachers in the use of alternative assessments. The EPSS is being designed to assist teachers at all levels of the assessment process covering a range of subject domains and grade levels. Among the many activities involved in using alternative assessments, this EPSS is being structured to support teachers in the design, selection, implementation, evaluation, or administration of a variety of performance-based assessments coming under the rubric of alternative assessment. This paper presents critical design issues that developers may encounter in developing an EPSS for professionals. The Alternative Assessment Resource Center for Teachers (AARCT) will be presented as a case study to illuminate the issues discussed. The design issues are organized into three main categories outlining the development processes utilized: (1) Front-end Analysis; (2) Development; and (3) Evaluation and Research.

Front-end Analysis

Matching techniques with types of problems

Developers of EPSSs are currently using a combination of traditional instructional systems development (ISD) procedures, software engineering principles, human factors principles, and a great wealth of intuition based on experience about what makes sense and seems to work best (Gery, 1995; Witt & Wager, 1994). Because different types of professions require different sets of skills (e.g., computer technicians, customer service representatives, business managers, physicians, teachers), developers of EPSSs are being asked to support a range of job-related skills from procedural tasks to complex problem solving. While procedural-type tasks lend themselves nicely to traditional ISD front-end analysis procedures such as task analysis or procedural analysis, other tasks involving problem solving of ill-defined, complex problems might be better analyzed, and subsequently better supported, using other methods that examine how people in these contexts solve problems and make decisions in which there is usually no one right answer.

Ill-defined, complex problems that may require the expertise of an experienced problem-solver in a particular domain (or work environment) are difficult to adequately train people for using the traditional ISD approach. This is because for many ill-structured or ill-defined problems, heuristics rather than algorithms are required to achieve ends (Tripp & Bichelmeyer, 1990). A heuristic can be thought of as a "rule of thumb" method for solving problems. It is not considered to be a guaranteed method for solving the problem, but is usually faster and more tractable than the correct algorithm (Ashcraft, 1989). Heuristics are just one example of the repertoire of reasoning tools expert problem solvers have as a result of experience in a particular domain. Documenting and understanding the expert's reasoning processes as he or she reviews past experiences to solve similar or novel problems can be a powerful method for learning how an expert solves problem in a performance context.

Case-based Reasoning for Analyzing Complex Problems

In the absence of relevant task-specific procedures, teachers often rely on instructional principles derived from their experiences as teachers, or other teachers in the case of teachers new to the profession, to help guide them in the decision-making process. This may also involve adapting examples from support materials such as teacher handbooks, textbooks, videotapes, computer databases, or other types of curriculum resource materials. Similar to other types of professionals (e.g., physicians, attorneys, business executives, etc.) teachers become highly skilled at adapting relevant examples (i.e., instructional plans) from one situation to fit a new instructional problem that is related in some aspect. These types of problem solving and decision making strategies indicate, as some researchers (Kolodner, 1992; Riesbeck & Schank, 1989) have found, that experience plays an important role in both the development of domain-specific problem solving skills as well as the refinement of those skills when novel problems are encountered. The process of examining previous cases to solve similar problems is referred to as Case-based Reasoning (CBR).

Developers of the Alternative Assessment Resource Center for Teachers found that a case-based reasoning approach was the best method for analyzing the extensive and complex nature of issues involved in conducting alternative assessments. A traditional task analysis could not easily absorb the vast array of knowledge, skills, and expertise that teachers brought to bear when solving problems and making decisions that usually involved many instructional variables. For example, portfolio assessment requires the evaluation of a broad spectrum of information for making important decisions about a student's work. The teacher must take into account a variety of considerations relating to each student's achievement. A task analysis used to understand this process may uncover some key steps involved in the process, however, much of the context-specific problem solving knowledge utilized by experienced teachers may not be revealed. As Laffey (1995) notes in the development of an EPSS for technicians at Apple Computer, "the (employees) could easily see how the use of cases in (the EPSS) mirrored their own informal strategies for solving hard problems. The (employees) saw value in having access to a rich and broad set of real experiences and in the processes that keep the set of experiences current" (Laffey, 1995). Examining cases that are set in the context of the work environment that is to be supported can provide developers of EPSSs a powerful method for better understanding much of the knowledge, both explicit and implicit, that professionals utilize in performing their jobs.

Case Study Approach

It became clear to developers during the initial analysis phase of trying to identify key tasks related to conducting alternative assessments that much of the knowledge and skills that should be supported were actually evident in many of the general principles used by experienced teachers. Therefore, the decision was made to focus on identifying a range of assessment examples that represented many of the key principles guiding teachers in the various processes. Once many of these examples were identified, a process for documenting the cases using a multimedia format was determined. The use of multimedia case studies as a framework to both gather data and structure support information was decided to be the primary development strategy for collecting and organizing other support materials within the system.

Data Collection

A systematic data collection tool was developed for focusing efforts on how to best document cases when visiting the schools. Because time is always at a high premium for teachers, decisions such as what kinds of media elements to be gathered were made prior to arriving at the school by using the data collection worksheet. Developers conducted pre-interviews with teachers to get an idea of the types of elements that should be highlighted in cases, as well as to prepare teachers for videotaped interviews about their role in the assessments. This methodology for collecting a source of rich data for multimedia case studies proved to be helpful in maximizing limited time constraints.

Development Issues and Strategies

Rapid Prototyping Methodology

The developers of AARCT took the position from the beginning that developing an EPSS for teachers had to be a participatory design process involving the teachers if the product was ever going to be anything that teachers would really use. Also, to achieve a high level of usability there would also have to be a development process that afforded considerable amounts of iteration involving conceptualizing, developing, testing, and revising. This iterative design process which is common in software engineering is referred to as prototyping. When the process of constructing prototypes is accelerated, so that the time from beginning a prototype to evaluating user interaction with it is short enough to allow time for substantial changes to the product, then this process is usually termed rapid prototyping (Hix & Hartson, 1993).

A rapid prototyping methodology was chosen to afford developers the creativity and flexibility to "get it right" referring to desired usability standards. While the idea of creating the "perfect" interface may be unrealistic, the use of software engineering principles such as prototyping and usability testing are ways to increase the likelihood that users will be able to use the system successfully. This approach appears to be well supported in the EPSS literature as mostly anecdotal evidence. As Gery (1995) points out, "few (designers) are guided by a set of integrated and fully articulated design principles. Many innovations are the result of individual or team creativity and iterative design employing rapid prototyping coupled with ongoing usability and performance testing".

The first prototype proved to be a valuable lesson in usability testing. Developers constructed a working prototype using an authoring system that allowed them to test information/knowledge organization and navigational strategies (See Figure 1). A school building metaphor was used as an interface design. Also accompanying this were a panel of menu buttons that forced the user to delimit choices based on assessment type, grade level, and content/subject area. A focus group of fifteen school teachers viewed a demonstration of the system that had a few example pieces hard linked to one of the cases. After going through the initial demonstration sequence, the developers were soon made aware by a majority of the teachers that the delimiting strategy of narrowing down choices based on pre-defined categories was not only counter-intuitive but represented antithetical ways of thinking about instruction. Most of these teachers taught interdisciplinary units that combined topics from a wide variety of content domains, therefore, the navigation structure did not support the way they thought about instruction (or assessment). In addition, some of the representative teachers taught more than one grade level and did not like the "arbitrary selection" of grade levels. This could be thought of as a discrepancy between the developers' mental models of this domain and the teachers' mental models. A mental model, simply put, is the learner's naturally evolving model, or internal representation, which summarizes his or her own knowledge about a process or thing, often taking the form of an informal theory (Norman, 1983). Mental models are representations that are active while solving a particular problem and that provide the workspace for inference and mental operations (Halford, 1993). The teachers asked for an interface that reflected their mental models -- an interdisciplinary approach to content domains, grade levels, and assessment strategies. Their mental models of how the interface should look reflected how they think about their jobs and the tasks involved. This information was lacking for the developers and became evident in their interface design.

In summary, it is useful to first interview intended users to get an idea at some level about how they think about a task or performance context. Then, start with an idea and test its conceptual validity with a sample of intended users before launching into any high-level screen design. By-passing this sequence may only prove that you have a different mental model about a type of work environment and its associative tasks than someone who actually works in that particular performance context (See Carroll & Olson, 1984).

Formative Experimentation

A strategy that works well with the goals of rapid prototyping to include the intended users in the design process is called formative experimentation. Newman (1990) describes formative experiments as a type of research/design method that "sets a pedagogical goal and finds out what it takes in terms of materials, organization, or changes in the technology to reach the goal". In this way design, formative evaluation, and research are all tied together and directed toward reaching a goal. In this case, developers of the AARCT are examining ways in which to best support teachers from a variety of backgrounds and levels of expertise in the use of alternative assessments. It is believed to really achieve this that developers will need to work closely with teachers using prototypes of the EPSS to see how they are using it, if at all.

As others have suggested (Goodrum et al., 1993), by examining the socio-technical aspects of technology in the workplace, it becomes important to recognize that technology is only one critical, highly interactive, and interdependent component of the whole organization or setting. When considering the use of an EPSS to support performance in a work environment it is necessary to consider all of the elements and methods that make up the way in which work is conducted. Developers of EPSS need to examine the social, cultural, and organizational aspects that affect the way in which work is performed in the work environment. By using methods like formative experimentation, developer can "test" and "tweak" systems to take into account many of these factors influencing work.

The developers of the AARCT are interested in seeing not only how teachers are using the system to support them in the use of alternative assessments, but they are also interested in seeing how teachers will restructure their work as a result of having this system available. Will they make it part of their normal planning time to utilize the system? Will they use the system to help manage the tasks involved in alternative assessments? Will they share information learned about the system with other teachers? Will they find it desirable to update the system periodically to include new cases? These are just some of the questions that the developers of the AARCT will examine as the system is tested in a few of the schools participating in the project. Many other questions will be posed throughout the formative experimentation process to see how the goal of supporting teachers in these tasks can best be achieved with the EPSS.

Conclusion

Developers of EPSSs should examine a variety of development strategies in an attempt to find out what best works for their situation. The developers of the AARCT are utilizing principles found in instructional systems, software engineering, performance technology, ano formative experimentation to develop an effective EPSS for teachers. There are, of course, many other strategies that could prove to be valuable as well in this and in similar endeavors. What is important to keep in mind, however, is that by making usability a goal and focusing on the social, cultural and organizational factors that influence how work is performed by professionals in a specific work context, developers increase the likelihood of developing effective EPSSs that support the people and the organization in meeting their goals.

References

Ashcraft, M. H. (1989). Human memory and cognition. Cleveland, OH: HarperCollins Publishers.

Brush, T., Knapczyk, D., & Hubbard, L. (1993). Developing a collaborative performance support system for practicing teachers. Educational Technology, 33(11), 39-45.

Carroll, J. M., & Olson. J. R. (Eds.). (1984, May 15-16.). Mental models in human-computer interaction: Research issues about the user of software knows. Workshop on Software Human Factors: Users' Mental Models, National Academy of Science, Washington, DC.

Collis, B., & Verwijs, C. (1995). A human approach to electronic performance and learning support systems: Hybrid EPSSs. Educational Technology, 35(1), 5-21.

Geber, B. (1990). Goodbye classrooms (redux). Training, 27(1), 27-35.

Gery, G. (1991). Electronic performance support systems. Tolland, MA: Gery Associates.

Gery, G. (1995). Attributes and behaviors of performance-centered systems. Performance Improvement Quarterly, 8(1), 47-93.

Goodrum, D. A., Dorsey, L. T., & Schwen, T. M. (1993). Defining and building an enriched learning and information environment. Educational Technology, 33(11), 10-20.

Halford, G. S. (1993). Children's understanding: The development of mental models. Hillsdale, NJ: Lawrence Erlbaum Associates.

Hix, D., & Hartson, H. R. (1993). Developing user interfaces: Ensuring usability through product & process. New York: John Wiley & Sons.

Kolodner, J. L. (1992). An introduction to case-based reasoning. Artificial Intelligence, 6, 3-34.

Laffey, J. (1995). Dynamism in electronic performance support systems. Performance Improvement Quarterly, 8(1), 31-46.

Law, M. P. (1994, February). Electronic performance support systems: Cognitive training wheels for the acquisition of skilled performance. Paper presented at the Annual meeting of the Association for the Development of Computer-Based Instructional Systems, Nashville, TN.

Newman, D. (1990). Opportunities for research on the organizational impact of school computers. Educational Researcher, 19(3), 8-13.

Norman, D. A. (1983). Some observations on mental models. In D. Gentner & A. L. Stevens (Eds.), Mental models (pp. 199-324). Hillsdale, NJ: Lawrence Erlbaum Associates.

Riesbeck, C. K., & Schank, R. S. (1989). Inside case-based reasoning. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Schwen, T., Goodrum, D., & Dorsey, L. (1993). On the design of an enriched learning and information environment. Educational Technology, 33(11), 5-9.

Scribner, S., & Sachs, P. (1991). Knowledge acquisition at work. (Technical Paper No. 2). New York: Institute on Education and the Economy Teachers College, Columbia University.

Tripp, S., & Bichelmeyer, B. (1990). Rapid prototyping: An alternative instructional design strategy. Educational Technology Research & Development, 38(1), 31-44.

Witt, L. C., & Wager, W. (1994). A comparison of instructional systems design and electronic performance support systems design. Educational Technology, 34(6), 20-24.

Figure Caption

Figure 1. Alternative Assessment Resource Center for Teachers (EPSS) Prototype 1.0.