1999

Reviewing Software as a Means of Enhancing Instruction

Marcus D. Childress
Old Dominion University

Guang-Lea Lee
Old Dominion University

Gregory P. Sherman

Follow this and additional works at: https://digitalcommons.odu.edu/teachinglearning_fac_pubs

Part of the Computer Sciences Commons, Early Childhood Education Commons, and the Teacher Education and Professional Development Commons

Reviewing Software as a Means of Enhancing Instruction

MARCUS D. CHILDRESS AND GUANG-LEA LEE
Darden College of Education 244 Education Building Old Dominion University Norfolk, VA 23529, USA
mchildre@odu.edu
LxLee@odu.edu

GREGORY P. SHERMAN
220 War Memorial Hall
Virginia Polytechnic Institute & State University
Blacksburg, VA 24061, USA
shermang@vt.edu

A software review procedure developed by the authors is described. The procedure centers around a form that extends the functionality of traditional software evaluation forms by enhancing the use of the computer software in the early childhood classroom. The form encourages teachers to discover ways a piece of software can be integrated across disciplines and used as an anchor for instruction. Users of the review form are also encouraged to examine ways the software motivates the user to remain engaged in its use. The majority of the review form emphasizes the important role of the teacher in identifying outcomes or skills, which may be acquired while using the software. Outcomes or skills are divided into nine areas of development: (a) physical development, (b) social/emotional development, (c) language development, (d) math/science development, (e) problem solving development, (f) self-esteem/confidence development, (g) aesthetic development, (h) multicultural awareness, and (i) creativity development. After addressing the outcomes in the nine areas of development, the reviewer creates activities, which can be applied to those areas. The review form was developed as an

>aid to educators in using software, which is age appropriate
>and individually appropriate.

Teachers of young children can rely on many sources to help them se-
lect potentially effective computer software. Numerous journals, catalogs,
and magazines devote sections or entire issues to software reviews. Soft-
ware evaluation forms serve as common tools, aiding teachers in the selec-
tion of appropriate software for their students. Haughland and Shade (1994)
provide an excellent example of an evaluation checklist, which identifies
three main areas to consider. These areas include child features, teacher fea-
tures, and technical features. In the child features checklist, terms such as
"active learning," "child-controlled interaction," "possible experimenta-
tion," "educational value," and "concrete representations" are used. Teacher
features include terms and criteria such as "childproof," "curriculum con-
gruence," represents differing ability and ages," and "understandable users'
manual" are used. The technical features area addresses items such as real-
ism, aesthetics, music, sound effects, graphics, and installation. By going
through the software selection process with such an evaluation form, teach-
ers can select a useful and developmentally appropriate piece of software,
which they can then attempt to integrate into their classroom.

What happens when teachers actually put specific software into use? A
typical scenario includes teachers placing the students at computers with the
software, expecting that the computer program will magically create a com-
plete environment suitable for learning. Unfortunately, teachers often give
little thought to the intended instructional purpose of the software, learning
objectives supported by the software, activities which may be linked to
these objectives, or where in the instructional process the software falls.
The purpose of this paper is to present a software evaluation and review
form that can be used in conjunction with more traditional computer soft-
ware evaluation criteria so that educators might address the more difficult
concerns related to how a particular software title might be most appropri-
ately utilized.

THE SOFTWARE REVIEW FORM

The National Association for the Education of Young Children
(NAEYC) "Position Statement on Technology and Young Children–Ages
Three through Eight" stresses the importance of the teacher in computer

software selection (1996). Teachers are to take into consideration whether the software is age appropriate, individually appropriate, and culturally appropriate. Furthermore, technology should be integrated into the classroom's daily routine across subject matter areas. Figure 1 illustrates a software review form developed by the authors to address these often neglected issues. Four examples of forms completed by teachers can also be found in the appendix of this book. Used in conjunction with a software evaluation form, such as Haughland and Shade's (1994), this review instrument can encourage teachers to look beyond the selection of software and investigate actual classroom implementation.

CONVENTIONAL REVIEW ITEMS AND SOFTWARE CATEGORIES

The software review form begins with conventional evaluation/review items such as the (a) title, (b) publisher, (c) year, and (d) brief description of the software. However, items five through twelve require more explanation. Item five provides a checklist for different software types. Traditional software types, such as drill and practice, tutorial, and game remain on the list. However, some nontraditional types may need further explanation. Creation software, most-simply defined, allows the user to create or make something. Simulation software allows the user to make decisions in the development and subsequent operation of a simulated environment or situation. Simulations often try to replicate real-world environments and usually represent "problem-solving" scenarios. Unlike simulation software, situation exploration software does not allow the user to make decisions regarding the development of a simulated environment; however, the user can freely explore within a simulated environment or situation. This type of software is also generally "problem-solving" in nature. Reference/Exploration software allows the user to investigate and access reference-type information. Game software engages the user in competition, usually for the sake of entertainment. Many categories of educational software take advantage of the motivational features of the game format. Electronic books give the user a chance to learn and explore while reading. These CD-ROM books often use multiple languages, animation, sound effects and music to engage the learner.

Most software titles include features that could be categorized into more than one group. When using the form, the reviewer should check all categories which apply to the software.

GRADE LEVEL AND/OR SUBJECT MATTER

In item six, the reviewer lists the appropriate grade level and/or subject matter for the software. Although most software is recommended for a specific age or grade level, many software titles can be applied to various ages and grade levels. In addition, most software titles are recommended for specific content areas or disciplines. However, with a little creativity, teachers can use the software as a theme or "anchor" between several disciplines. Anchored instruction occurs when learning experiences are organized around a theme which serves as a conceptual thread throughout the curriculum (Risko, 1990). Through the use of anchored instruction, students can use the common theme to connect information and concepts from different disciplines. In anchored instruction, the teacher creates an environment which permits student exploration in problem solving in real-world situations, not just those from a single textbook (The Cognition and Technology Group at Vanderbilt, 1990). The use of an anchor aids the student in acquiring new information and organizing that information into existing conceptual frameworks (Bransford, Sherwood, Vye, & Reiser, 1986).

MOTIVATING CHARACTERISTICS

In item seven, the reviewer lists ways the program motivates the user to remain engaged. Motivation is a key component of learning. In order to learn, one must be motivated. Sources of motivation for learning include curiosity, achievement, and self-efficacy (Gagné & Driscoll, 1988). Motivating characteristics in software include humor, concreteness, animation, music, and rewards of some type. By completing item seven, the reviewer consciously examines the software for motivating features and, if necessary, addresses the motivational deficiencies of the software.

PREREQUISITE OUTCOMES OR SKILLS

In item eight, the reviewer lists outcomes or skills a student must already be proficient at performing in order to successfully use the software. The reviewer must consider the entry-level skills, which must already be possessed by the learner. One of a teacher's most crucial duties involves selecting developmentally appropriate instructional materials for his/her students. As Shade (1996) aptly states, "Just as how crayons are used depends

on whether children are given blank paper or coloring books, the use of a computer is determined by the developmental appropriateness of the software selected" (p.17). In completing item eight, the reviewer must take into consideration the appropriateness of the software in light of the students' current skill level.

DESIRED OUTCOMES OR SKILLS

In item nine, the reviewer identifies outcomes or skills attainable from or supported in some way by the software. The outcomes/skills, divided into nine areas of development, include: (a) physical development, (b) social/emotional development, (c) language development, (d) math/science development, (e) problem solving development, (f) self-esteem/confidence development, (g) aesthetic development, (h) multicultural awareness, and (i) creativity development. On the form, the reviewer lists desired outcomes or skills in each development area. Through observations of preservice and inservice teachers, it has become apparent that some teachers fail to understand what outcomes or skills can be developed from software. For example, an interactive electronic book, such as Broderbund's *Just Grandma and Me*, could easily act as an anchor in learning about subjects other than its most obvious intended use of language development. Although it is an electronic book, the software can be used to support the learning of skills associated with marine life, mathematics, social studies, and general problem-solving.

ACTIVITIES

In item 10, the reviewer lists activities in the areas of development in which a child could engage, while exploring the software. In this item, the teacher's creativity most often comes into play. Based upon the outcomes and skills identified in item nine, the reviewer describes activities which will maximize the probability that the targeted skills will be acquired. Because some educators may only think of the "recommended" uses for software implementation, they may not consider developing activities to address other outcomes. Item 10 can aid the teacher in the development of such activities.

OUTSTANDING FEATURES

In item 11, the reviewer lists characteristics of the software which can be described as "outstanding features." A high degree of individualization may be cited, or perhaps the use of animation or music may be characterized as outstanding. This item gives the reviewer the opportunity to list such outstanding features.

LIMITATIONS

In item 12, the reviewer lists any limitations encountered when using the software. Limitations might include attributes such as confusing navigation, lack of interactivity, or poor animation. This item gives the reviewer a chance to identify features of the software which might truly prohibit the program from being useful.

CONCLUSION

Once the review form is completed, the teacher/reviewer should be more aware of: (a) the ways a piece of software can motivate the learner, (b) skills needed before using the software, (c) outcomes or skills which can be acquired from the software, and (d) activities that address the nine areas of development. Having such a tool, the teacher may be better prepared to appropriately integrate the computer into the curriculum. As the NAEYC's "Position Statement on Technology and Young Children–Ages Three through Eight" states:

> Teachers should look for ways to use computers to support the development and learning that occur in other parts of the classroom and the development and learning that happen with computers in complement with activities off the computer. (National Association for the Education of Young Children, 1996, p. 2)

When asked about the review form's usefulness, teachers have responded favorably. Responses such as "When I think of a child learning a skill, I would not pick this CD. But, after using the form, I realize that a child can learn a lot from this program," suggest that using the form can promote the creative use of software which was primarily intended for entertainment. Many teachers found that the review form helps them to "come

up with outcomes, because it gives the areas of development." Although completing the form can be time-consuming, it enables educators to think of outcomes and activities for areas that they may have never considered. As one teacher responded, "When I previously looked at software, I did not really consider things such as self-esteem, social/emotional, or physical development." This review form can assist early childhood educators in attaining the NAEYC goal. Even the most popular and heralded software has limited value, without the proper instructional design and implementation by the teacher. This review form holds the potential to become an important tool in helping teachers develop instructionally sound objectives and activities for the classroom.

References

Bransford, J.D., Sherwood, R., Vye, N., & Reiser, J. (1986). Teaching thinking and problem solving: Research foundations. *American Psychologist*, *14*(10), 1078-1089.

Gagné, R.M., & Driscoll, M.P. (1988). *Essentials of learning for instruction*. Englewood Cliffs, NJ: Prentice Hall.

Haughland, S.W., & Shade, D.D. (1994). Software evaluation for children.

In J.L. Wright and D.D. Shade (Ed.), *Young children: Active learners in a technological age*. Washington, DC: NAEYC.

National Association for the Education of Young Children. (1996). NAEYC position statement: Technology and young children–ages three through eight. *Young Children*, *51*(6), 11-16.

Risko, V. (1990). Effects of videodisc macrocontexts on comprehension and composition of causally coherent stories. Paper presented at the Annual Meeting of the American Educational Research Association, Boston, MA, April 15-21, 1990. (ERIC Document 318 998)

Shade, D.D. (1996). Software evaluation. *Young Children*, *51*(6), 17-21.

The Cognition and Technology Group at Vanderbilt. (1990). Anchored instruction and its relationship to situated cognition. *Educational Researcher*, *19*(6), 2-10.