Demobank: a method of presenting just-in-time online learning

R. David Lewis
University of South Florida
Address for Correspondence:
Florida International University,
11200 S.W. 8th Street GL141,
Miami, Fl, 33199
lewisd@fiu.edu

Demobank: a method of presenting just-in-time online learning

Abstract

A demobank is a method of presenting web-based instruction for promoting procedure-based learning. This presentation technique is a well structured collection of web-based animated demonstrations (narrated demonstrations of computer-based procedures). A demobank allows the learner to view animated demonstrations individually or as a series. These demonstrations usually depict a group of computer-based procedures, which if properly constructed can act as a help system or as a "just-in-time" online workshop. Animated demonstrations are increasingly being used by a number of companies to deliver web-based training to their intended audiences, either potential clients or employees. This paper discusses learning given a demobank of animated worked examples and design considerations for this new presentation technique.

Animated worked examples?

Many people believe they must first perform a procedure to learn that procedure, sometimes repeatedly. However, one can learn how to perform a procedure just by studying examples (Sweller and Cooper, 1985). Certainly the correct performance of a procedure reinforces learning, but as Sweller and Cooper clearly demonstrate, performance is not truly necessary to learn a procedure. They found that learners that studied worked examples required significantly less time during skill acquisition, and that these learners made significantly fewer errors during problem solving. This paper builds upon Sweller's worked examples studies to produce a just-in-time learning environment which is generalizable to all computer based applications.

For decades researchers have known that graphics are necessary in instruction, because they support verbal communication (Bransford and Johnson, 1972). More recently it has been shown that learners using multimedia significantly out-performed learners using single media instruction (Brünken, Steinbacher, Plass, and Leutner, 2002; Mayer and Moreno, 1998; Mousavi, Low, and Sweller, 1995). Mousavi, Low and Sweller (1995) suggested that the simultaneous use of visual and verbal media effectively increases the learners' working memory capacity. Brünken et al. (2002) suggested that this is because multimedia makes use of the combined resources of both the visual and auditory subsystems of working memory.

Chandler and Sweller (1992) succinctly define worked examples as "a problem statement and the appropriate steps to solution (Chandler and Sweller, 1992, p. 233)." Animated demonstrations clearly fall under this category as they describe a problem and its solution in a series of steps. Because animated demonstrations are goal directed and procedure-based they act as animated worked examples.

Several researchers have studied the instructional effectiveness of animated demonstrations (Palmiter, 1991; Harrison, 1995; Lipps et al., 1998; Waterson & O'Malley, 1993). Like Sweller, these researchers, found that this form of instruction speeded skill acquisition, and that learners were faster and more accurate when learning procedural tasks. Sweller suggests this is because these learners concentrate on relevant problem states and the steps necessary to solve problems (Sweller et al., 1998). In addition performance may be improved because of the multimodal nature of this form of instruction.

Given such a useful form of instruction, how might we improve upon this presentation technique? Sweller suggests instructional designers should design instruction with the human cognitive architecture in mind (Sweller, 1993, 1999). The remainder of this paper explores how Sweller's statement can be applied to the design of a demobank of animated worked examples.

How learning occurs given animated worked examples

A distinction should be made between procedural and declarative knowledge. Declarative knowledge refers to mental representations of the outside world; whereas procedural knowledge refers to processes that operate on those mental representations (van Merriënboer, 1997). Declarative knowledge is primarily concept or language based whereas procedural knowledge is usually goal directed and aimed at "knowing how." Demobanks may describe declarative information to learners, but are primarily useful for teaching learners procedural knowledge. Given complex cognitive skills require both declarative and procedural knowledge (van Merriënboer, 1997), a demobank becomes an ideal medium for teaching these complicated skill sets.

Sweller and Cooper (1985) suggested worked examples were useful mainly in the initial stages of skill acquisition. Researchers have been describing the phases of skill acquisition for over forty years (Fitts, 1964; Anzai and Simon, 1979; Anderson, 1983). Anderson continued in this tradition when he described cognitive skill acquisition (Anderson, 1982). In Anderson's ACT framework (Anderson, 1976, 1983, 1993) the three stages of skill acquisition are the declarative stage, the knowledge compilation stage, and then finally a procedural stage

(Anderson, 1983). Since animated worked examples are most useful early in skill acquisition, this paper will strictly focus on learning during the declarative stage.

During the declarative stage, learners mentally rehearse procedures during a performance (Anderson, 1982). Anderson suggests it is even common to observe learners in this phase verbally rehearsing the steps of a procedure (Anderson, 1983). Performance during this first stage although laborious is often poor and error prone (Shiffrin and Dumais, 1981). This is because learners must first understand under what conditions a particular procedure is used. In addition they may also be considering the current problem state, the end goal and differences between the two (Ward and Sweller, 1990). So it is understandable that novice learners may have some difficulty.

Given Anderson's framework, the steps of a procedure are worked out according to procedure specific production rules. Production rules involve if then statements like the following:

IF the goal is to communicate with someone in another state

THEN dial the telephone.

A grouping of production rules is called a production. Productions are similar to what a behaviorist would have called a stimulus-response pair, but from a cognitive perspective (Anderson, 1983), because it takes into account decision making. So procedural knowledge according to Anderson's framework is initially represented verbally in working memory, and then later during many hours of practice converted into the fluid movements of an expert.

Chi et al. (1981) noticed that experts analyze problems differently than novices. These experts gather information to categorize problems usually according to schema specific rules (Chi et al, 1981). Once a problem is categorized, the expert may use a set of schema specified production rules to solve that particular problem (Chi et al, 1981). Chi and Anderson are working from complementary theory bases. Chi tends to place more emphasis on categorization. In summary, a problem schema is basically a set of production rules that guides a problem solver to first categorize a problem, and then to solve that problem, by following the production rules for that specific problem category.

Without the guidance of a demobank, novice learners may use weak methods like "means ends analysis to try to solve a problem (Owen and Sweller, 1985; Anderson, 1987). Being that a demobank of animated demonstrations is a collection of animated worked examples the literature on learning by studying examples will be described in this next section.

Studying worked examples versus means-ends analysis

Owen and Sweller, (1985) found that most novices try to solve problems by attempting to achieve a series of sub goals. This strategy is called "means ends analysis." Their results suggested that this strategy "places a heavy load on cognitive processing capacity (Owen and Sweller, 1985, p.272)." This is perhaps Sweller's first reference to cognitive load. After making this statement John Sweller went on to describe cognitive load theory (Sweller, 1988).

Sweller developed cognitive load theory as a means of explaining the performance of novices during schema acquisition. He proposed that during the initial stages of learning, learners are bombarded by information, which if not properly regulated, can quickly overload a learner's limited "working memory" and thus deteriorate performance (Sweller, 1988). Cognitive load theory contends that if instructional designers limit the amount and complexity of information impacting learners that this will promote more efficient schema acquisition and automation (Sweller, 1988, 1993, 1994).

Means-ends analysis is like trying to get through a maze by going in what seems to be the best direction at the moment. Learners using this strategy often make errors, given the maze analogy they may try the wrong path toward their eventual goal. Many flounder and may find themselves going in circles. Errors like these may cause the learner a great deal of frustration.

The use of worked-examples on the other hand, decreases cognitive load by presenting only enough information to learn how to solve a problem. In terms of the maze metaphor, learning with worked examples provides the learner with guidance that can shorten their route toward learning. In many cases learners learning this way as opposed to learning by traditional problem solving (via means ends analysis) required less time to study the examples and then later less time to solve problems (Sweller and Cooper, 1985). Ward and Sweller (1990) explain means ends analysis this way: "A heavy cognitive load is imposed because of the need to simultaneously consider and make decisions about the current problem state, the goal state, differences between states, and problem solving operators that can be used to reduce such differences (Ward and Sweller, 1990, p.3)."

Learners using means ends analysis, often make errors and waste time trying to solve problems, rather than using that time effectively to learn the underlying problem schema. Later because of these errors those using means ends analysis may have difficulty piecing together the underlying steps of the overall procedure.

Zhu and Simon (1987), later went on to replicate Sweller and Cooper's findings with a group of high schools students that were learning algebra and geometry. They were able to show that learners using worked examples could complete a 3 year curriculum, in only 2 years -- a full year ahead of their peers who had learned through a traditional problem solving curriculum. Zhu and Simon (1987) also found that this group of students scored slightly higher on national achievement tests as compared with their problem solving peers.

The results from these early studies (Sweller and Cooper, 1985; Cooper and Sweller, 1987, Zhu and Simon, 1987) are encouraging, but somewhat counter intuitive; and certainly they are not without their critics (e.g. Gabrys et al., 1993). This is because educators have had "learning by doing" ingrained in educational culture, since the time of Dewey (Kilpatrick, 1918). But Sweller's empirical results still suggest learners using worked examples learn to perform those skills quicker and with fewer errors (Sweller and Cooper, 1985). Why might this be the case?

Chi et al. (1981) compared experts and novices as they solved physics problems. They found that experts solve problems based upon solution mode. Novices on the other hand concentrated on the surface features of the problems. By presenting novices with worked examples instructional designers teach them to be more like experts. That is they are taught to categorize problems according to solution mode. This is as compared to traditional problem solving techniques which expect learners to grasp the means to a solution while being simultaneously burdened with the task of solving the problem. Traditional problem solving adds cognitive load to an already demanding situation.

While studying learners using worked examples, Chi et al. (1989) shed some additional light on the worked example effect. They suggested students learn via self-explanations. This supports Anderson framework of verbal explanation of problems during schema acquisition (1982). Chi et al. (1989) monitored students using think aloud protocols. They found that "poor" students do not generate sufficient self explanations. "Good" students on the other hand monitor their understanding and misunderstanding. In doing so they were better able to locate areas of difficulty and concentrate on those areas. Unfortunately "poor" students were unable to locate problem areas and in turn had difficulty when the time came to perform.

Practice of course is important after initial learning and indeed this is what Zhu and Simon (1987) did in their longitudinal study. They initially taught students using worked examples so that they would learn to categorize problems based upon solution mode. Then once learners could categorize problems effectively they provided them with practice solving problems. This allowed them to automate their skills (Schneider and Shiffrin, 1977; Shiffrin and Schneider, 1977) and promoted what Bloom later described as "automaticity (Bloom, 1986)."

Implications for demobank design

The problems an instructional designer can provide via a demo bank are diverse. As mentioned earlier this presentation technique is generalizable to all computer based instruction. An instructional designer must make several important decisions about how to present instruction via this form of instruction. Therefore this section will apply the preceding discussion to the practice of demobank design and provide a set of demobank design guidelines. But first an example will be provided to give some context to these guidelines.

A demobank was developed by the author to produce a series of web-based animated worked examples for university faculty (note figure 1). Our campus PeopleSoft implementation (PantherSoft) became the subject matter for this tutorial. The menu is designed to support the schema behind this software. For each learner action a menu item opens an animated worked example.

The initial implementation of this project became very useful as a tool to teach large numbers of people at their own convenience. Because this multimedia learning environment is posted on the web it is available 24 hours a day 7 days a week. Given that a demobank is posted on the web, learners can make use of it wherever they are as a just-in-time online workshop. In addition it is also available to learners for review, as they solve problems. The PantherSoft tutorial is just one example of a demobank, which can be generalized to form a general classification of tutorials. Several design issues were found to be apparent during the development of the above example. These are documented here and can serve as guidelines for future designers.

First and foremost it is very important for designers to work with experts in order to categorize problems by solution mode. Expert users can suggest how a task can be broken down into constituent sub skills. Given that designers arrange worked examples according to how experts view problems, learners will begin to view the software like experts.

Providing smaller segments will be beneficial to the learner in two ways. First it prevents working memory overload by decreasing the amount of information impacting the learner. In addition it will be easier on the learner's computer because smaller video segments require less bandwidth. Another bandwidth consideration is the use of audio. Audio does require more bandwidth, but resist the temptation to remove the audio from the presentation. As mentioned, earlier multimodal instruction (simultaneous use of visual and verbal media) has been shown to promote learning because it makes more effective use of a learners working memory (Mousavi, Low and Sweller, 1995).

As Gagne (1965) explained many years ago the sequence of instruction is very important. It would be useful to list animated demonstrations in order if possible. Be clear why these procedures matter and explain under what conditions a particular procedure is used and why the order or sequence of tasks is important. Perhaps consult the advice of an expert to get a better understanding of the sequence of tasks.

Remember Chandler and Sweller described a worked example as "a problem statement and the appropriate steps to solution (Chandler and Sweller, 1992, p. 233). Therefore it is important to explain the goal of the problem at hand. This confirms for the learner that they are watching the appropriate presentation. In addition explain the context of the current presentation as it relates to others in the demobank. This will provide the learner with the appropriate context.

Learners may wish to return to your tutorial on another occasion. For this reason it is important to make these demonstrations available as just-in-time instruction. Designers can suggest in the narration that learners review certain segments in the presentation. So suggest repetition and practice. Demobanks are a good way to present procedures to the novice, but without practice they will remain novices. The easiest way to make these demonstrations available as just-in-time instruction is by publishing them as a web site, complete with a menu of tasks based on the problem schema.

The development of animated demonstrations presents a different set of problems from most video production. During most video production, camera operators capture the simultaneous movement and audio of actors. This typically occurs through many takes that can later be seamlessly edited together to produce one continuous stream of video. However this is not as easily accomplished with screen capture software. Even though it is possible to record audio and video simultaneously, this is not suggested. Screen capture software products capture on-screen motion but it is suggested that this is only be integrated with audio segments in post production.

Therefore the most efficient way to produce animated demonstrations is to begin with the narration. Narration begins as a well written script. Spend time carefully considering your instructional message for here is where much of the instruction takes place. Once you have your script ready, have your voice talent practice the script once or twice before recording the narration. Narration can then be recorded as digital audio files in a computer. Several takes may be necessary to record the complete script. This audio file can then be edited to produce one continuous error free file. This narration file can then be played back to cue someone as they capture the video component of demonstration with screen capture software (e.g. Techsmith Camtasia or Macromedia Captivate). This should also be rehearsed several times before making a video capture. Next a developer can synchronize the audio and video tracks to produce a fluid animated demonstration. Finally these movies should be saved as a steaming file format, most developers prefer Macromedia's Flash file format because of its ubiquity and its cross-platform compatibility. These movie files should be displayed on a web page one at a time. In addition each page will contain a schema-based menu system that allows the learner to view files one at a time at their own pace.

Summary

Sweller suggests instructional designers should design instruction with the human cognitive architecture in mind (Sweller, 1993, 1999). Given schema theory, an ideal demobank would exactly mirror an expert's problem schema. One may think of a demobank of animated worked examples as an externalized problem schema. Given this conception of a schema, presenting a demobank to a novice learner, can be thought of as "schema sharing." This is because this presentation technique allows the designer to group animated examples according an expert schema. Learners can use this built in organization to constructively group procedures into problem categories that can be used to solve problems according to procedures outlined by the animated worked examples. Finally once a schema has been incorporated into the learner's long term memory, perhaps through the performance of the demonstrated procedure, that schema is successfully shared with a learner.

References

Anderson, J.R. (1976). *Language, Memory and Thought*. Hillsdale, NJ: Erlbaum Associates.

Anderson, J.R. (1995). *Cognitive psychology and its implications*. (4th ed.) New York: W.H. Freeman and Company

Chi, M. T. H., Feltovich, P. , & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5, 121-152

Bartlett, F.C. (1995). *Remembering: An Experimental and Social Study*. Cambridge: Cambridge University Press.(Original work published 1932)

Bobrow, D. G., & Norman, D. A. (1975). Some principles of memory schemata. In D.G. Bobrow & A. M. Collins (Eds.), *Representation and understanding: Studies in cognitive science*. New York : Academic Press

Bransford, J. D., & Johnson, M. K. (1972). Contextual prerequisites for understanding: Some investigations of comprehension and recall. *Journal of Verbal Learning and Verbal Behavior*, 11, 717-726.

Fitts, P. M. (1964). Perceptual-motor skill learning. In A. W. Melton (Eds.), *Categories of human learning*. New York: Academic Press.

Harrison, S. M. (1995). A Comparison of Still, Animated, or Nonillustrated On-Line Help with Written or Spoken Instructions in a Graphical User Interface in Proceedings of CHI 95, Association of Computing Machinery retrieved October 14, 2003 from http://www1.acm.org/sigs/sigchi/chi95/Electronic/documnts/papers/smh_bdy.htm

Kilpatrick, W. H. (1918). The project method. *Teachers College Record*. 19, p 319-335

Mayer. R & Moreno R. (1998). A split attention effect in multimedia learning: evidence for dual processing systems in working memory. J*ournal of Educational Psychology*. 90, 2, 312 320.

Neves, D. M., & Anderson, J. R. (1981). Knowledge compilation: Mechanisms for the automatization of cognitive skills. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition*. Hillsdale, NJ: Erlbaum.

Shiffrin, R. M. & Dumais, S. T. (1981). The development of automatism. In Anderson, John (Ed.), *Cognitive Skills and Their Acquisition*, 111-140. Hillsdale, NJ: Erlbaum.

van Merriënboer, J.J.G. (1997). Training complex cognitive skills. A Four Component Instructional Design Model for Technical Training. Educational Technology Publications, Englewood Cliffs, NJ.

Figure 1. PantherSoft Student Administration Demobank