Developer Intent versus Instructor Delivery in Program Implementation

Cecelia L. López and
Howard J. Sullivan
Psychology in Education
Arizona State University
and
Sandra L. Weber
College of Engineering and Applied Sciences
Arizona State University

Abstract. Observation of the field testing of an instructional program for undergraduate engineering students revealed that the instructor did not collect or grade the student homework exercises as recommended in the instructor guide. Therefore, an experimental study was designed to investigate whether instructor delivery of the program as recommended would yield higher student achievement.

The study was conducted for six class periods with 19 students in the Fall semester and replicated with 14 students and a second instructor in the Spring. Homework exercises were collected, scored, and returned for the experimental group, as recommended in the instructor guide. The exercises were not collected for the control group, thus reflecting the procedure used by the instructor in the field test.

Posttest scores and grades were significantly higher for the experimental group. The test scores and grades by treatment were also consistent across the two semesters and two instructors. Self-report data revealed that experimental subjects spent significantly more time on the exercises.

Questioning of the instructor revealed that he did not collect or grade the exercises because of the amount of time it required. The instructional designer placed more responsibility for student achievement on herself and the instructor, whereas the instructor placed more responsibility on the students. The differing opinions of the instructional designer and the course instructor are discussed as they relate to the design and delivery of instructional programs.

Instructors often implement systematically designed instructional programs in a manner different from that intended by the developers (Burkman, 1987). Variations between developer intent and instructor delivery of instructional design products have been documented in a variety of settings, including schools (Branson, 1988) and the military (Branson, 1981; McCombs, Back, & West, 1984).

Stake (1967) noted that the discrepancy between what was intended and what actually happens in instruction may have a significant effect on student achievement. Hall and Loucks (1977) suggest that many nonsignificant findings reported in evaluation studies may actually be the result of erroneously assuming that the product is being used as intended by the experimental group. They argue that it is critical to have first-hand documentation that the product is, in fact, being implemented as intended. Because so much can occur during implementation that can alter the intended use of the product,

Instructors often implement systematically designed instructional programs in a manner different from that intended by the developers....

Durzo and Florini (1984) stress that it is important to determine if the product was used as planned and to assess how what was done affected performance outcomes.

The present development and research effort involved a computer-based instructional program for university students majoring in construction engineering. The instructional program, entitled *DOT Literacy* (Weber & López, 1987), was originally developed by an instructional designer and a civil engineering subject matter expert (SME), using systematic instructional development procedures derived from the works of Sullivan and Higgins (1983) and Gagné and Briggs (1979). Its overall purpose was to teach students the use of the IBM disk operating system (DOS), version 3.1.

The instructor for the field test of the DOS program did not, in fact, follow certain key procedures specified in the instructor guide. The guide called for the instructor to assign homework exercises in the student book and to collect and grade them. Questioned after the field test, the instructor cited lack of time as his reason for not collecting and grading the exercises.

The designer suspected that the instructor's failure to follow the procedures for homework exercises as described in the instructor guide resulted in lower student achievement. Therefore, she designed and conducted an experimental research study after implementation of the program to determine whether using it as designed would yield higher achievement.

The study involved two treatments administered as part of the regular instruction in the university engineering course in which the DOS program was used. Under the experimental treatment, the homework exercises were collected and graded as directed in the instructor guide. Under the control condition, the instructor taught the program as in the field test, without collecting and grading the exercises.

The study was conducted across two semesters to yield additional data because the course enrollment was relatively small—15 to 20 students per semester. Both an experimental and a control group were employed during each semester. One instructor taught the program in the fall and another instructor in the spring in order to provide data on the replicability and generalizability of the findings.

The DOS Program

The DOS program was designed as part of a course for construction engineering majors entitled "Microcomputers for Constructors." The program consists of four modules, one for each of its four instructional objectives listed below in abridged form.

1. Use basic DOS commands (e.g., DISKCOPY, FORMAT, and SYS)
2. Use advanced DOS commands (e.g., BACKUP, RESTORE, and CHKDSK)
3. Create and manipulate fixed disk subdirectories (e.g., MD, PATH, TREE)
4. Generate batch files, using batch file commands (e.g., ECHO, %n, GOTO) and DOS commands.

The program requires approximately seven hours of class time to complete. Homework assignments take several additional hours.

The format for each of the four modules follows a similar sequence. The instructor introduces the module and previews its content for the students. Basic information is presented both by the instructor and in the student book. The instruction for each module in the student book incorporates 15–20 practice questions with feedback, followed by a review of the purpose and syntax for each command. A homework exercise on the computer is included at the end of each module. The instructor guide directs the instructor to assign each homework exercise at its appropriate point. The program also includes a 40-item constructed-response posttest consisting of 10 items covering the instructional objective for each module.

The DOS program was field tested under regular classroom conditions in the spring of 1987 in the engineering course for which it was developed. The mean percentage score of only 5% on the pretest, a 40-item measure in a form equivalent to the posttest, revealed a definite need for the program. The mean posttest score for the field test was 61%. Observation of the field test by the instructional designer revealed that the field test instructor, who was also the regular course instructor, assigned the homework exercise for each module but did not collect and grade the exercises as described in the instructor guide.

Both the SME and the course instructor were pleased with the field test results and decided to incorporate the DOS program into their course. The instructional designer, however, hoped to improve student performance and identified use by the instructor of the specified homework procedures as a possible means of doing so. She therefore designed an experimental study to investigate whether the instructor's use of these procedures would, in fact, yield higher student achievement. The DOS program was incorporated into the regular "Microcomputers for Constructors" course beginning in the fall of 1987, and this course provided a natural setting for the designer to conduct the investigation.

. . . it is important to determine if the product was used as planned and to assess how what was done affected performance outcomes.

Method

Participants

Subjects were 33 undergraduate Construction Engineering majors enrolled in the "Microcomputers for Constructors" course in the School of Engineering at Arizona State University during the fall of 1987 ($N = 19$) and spring of 1988 ($N = 14$) semesters. Both the regular course instructor and the SME are Associate Professors and licensed Professional Civil Engineers.

. . . delivery of instruction as described in the instructor guide yielded significantly higher student posttest scores and grades. . . .

Materials

The instructional program was the previously described DOS literacy program on the disk operating system for the IBM microcomputer. The program was revised on the basis of field test observations and data prior to its final printing for use beginning in the fall of 1987.

Both the experimental treatment, in which the homework exercises were collected and scored, and the control condition, in which the exercises were not collected and scored, were employed each semester on a between-subjects basis. For each semester, subjects were assigned randomly to the two treatment groups. The regular course instructor taught the program during the fall semester. The SME substituted for the regular instructor for the DOS portion of the course during the spring semester. Classes met twice a week for an hour and 20 minutes, and instruction for the program took five class sessions.

The classes were informed that, as part of an experiment, the instructor would collect and grade the computer exercises of certain students, who were then identified by name, but not of the other students. The students whose work was to be collected and graded turned in printouts of their exercises at the beginning of the class period following each assignment. The experimenter scored the exercises for the instructor and wrote brief comments and grades on them. Exercises were not collected from the control students. The experimental subjects' exercises were returned to them at the next class session after they were submitted. The answers for the exercises were distributed to all students, both experimental and control, after the experimental group's work was returned. Answers to the exercises were incorporated into the student text in the field test, but were removed from the revised version of the text on the recommendation of the instructor and several students.

The posttest and an attitude questionnaire were administered to all subjects during the sixth class period.

Criterion Measures

The primary criterion measure was the 40-item constructed-response posttest which covered the same instructional objectives as the practice questions and the computer exercises. Posttest inter-item reliability calculated with Kuder-Richardson formula 20 was .88.

The grades which instructors assigned to students for the program provided the basis for a second criterion measure. The instructor for each semester assigned the grades from test scores without knowledge of the subjects' names and, consequently, of their particular treatment group. Grades supplemented the posttest as a criterion measure by providing a correlated, but more global, measure of the importance of any observed differences in test performance.

The 10-item attitude questionnaire assessed students' attitudes and perceptions related to the assignments. The items dealt with matters such as students' willingness to do computer exercises on other topics, their preferences related to having the exercises graded, the time they spent on the exercises, and how important they thought the exercises were.

Design and Data Analysis

The experimental design was a randomized posttest-only control group design (Campbell & Stanley, 1963, Design 6). Posttest achievement was analyzed by t test, and questionnaire items were analyzed individually by chi-square.

Posttest scores were very similar within treatments across the two semesters and instructors, and t tests of the within-treatment differences across semesters revealed that they were not significant for either treatment. Data were therefore pooled across semesters for the between-treatment analyses.

Results

Posttest Performance

Table 1 shows that the mean posttest scores were 25.83 (65%) for the experimental group and 20.88 (52%) for the control group. This difference was statistically significant, $t\,(1,31) = 2.08$, $p < .05$.

The table also shows that the greatest between-treatment difference within a module occurred in Module 4 (generating batch files), which requires the most complex student performance. Mean scores for this module were 5.63 (56%) for the experimental group and 3.35 (34%) for control, also a statisti-

TABLE 1
Means and Standard Deviations of Test Scores by Treatment and Time of Experiment

	Posttest Scores	EXPERIMENTAL Fall 1987	EXPERIMENTAL Spring 1988	EXPERIMENTAL Total	CONTROL Fall 1987	CONTROL Spring 1988	CONTROL Total	*Experimental-Control Difference* (E-C)
Module 1	M	7.11	7.14	7.13	6.50	7.00	6.71	.42
	SD	1.27	2.41	1.78	1.65	2.16	1.83	
Module 2	M	6.33	6.43	6.38	5.40	5.71	5.53	.85
	SD	1.50	1.81	1.59	1.96	2.75	2.24	
Module 3	M	6.89	6.43	6.69	5.40	5.14	5.29	1.40
	SD	2.76	2.88	2.73	1.78	3.08	2.31	
Module 4	M	5.11	6.29	5.63	3.10	3.71	3.35	2.28
	SD	1.76	2.98	2.36	2.47	1.98	2.23	
Total	M	25.44	26.29	25.83	20.40	21.57	20.88	4.95
	SD	4.64	8.08	6.15	6.47	9.05	7.39	

cally significant difference, t (1,31) = 2.84, $p < .01$. The experimental group also scored higher than the control group on each of the other three modules, but these scores did not differ significantly between treatments.

Grades

The grade distribution for the program summed across the two semesters is as follows:

Experimental	*Control*
A: 4	A: 2
B: 8	B: 4
C: 3	C: 6
D: 1	D: 4
E: 0	E: 1

These grades convert to an overall GPA of 2.94 for the experimental group and 2.12 for the control group. A t test revealed that the difference was statistically significant, t (1,31) = 2.37, $p < .05$. Grades within treatments were similar across the two semesters: 3.0 for experimental subjects and 2.10 for control subjects in the fall, and 2.86 for experimental subjects and 2.14 for control subjects in the spring.

Student Attitudes

Student responses to the 10-item questionnaire revealed significant differences favoring the experimental group over control subjects on two items, time on task and perceived level of task difficulty. Sixty-three percent (10 of 16) of the experimental subjects and 24% (4 of 17) of control subjects reported spending from three to five hours or more on the computer exercises for each module, χ^2 (1,31) = 5.12, $p < .05$. The remaining subjects reported spending less time on the exercises. Sixty-five percent (13 of 20) of the experimental subjects and 41% (7 of 17) of control perceived the computer assignments as difficult, χ^2 (1,31) = 5.54, $p <.02$. The attitudes of the experimental group were more positive, but not significantly so, on six of the remaining eight questionnaire items.

Discussion

The overall results of the study reveal that delivery of instruction as described in the instructor guide yielded significantly higher student posttest scores and grades in the experimental treatment. The critical instructor behavior in producing this effect was the collecting and grading of assigned homework. Student reports indicated that the same instructor behavior also resulted in greater student time spent on the homework task.

The most likely dynamics underlying the performance differences relate to the fact that experimental subjects knew that their work would be collected and graded, whereas control subjects knew that theirs would not. Consequently, experimental subjects spent more time on the exercises and thereby had more practice on the instructional objectives covered in them and assessed on the posttest. Both practice and time on task have been found to improve learner performance (Craik & Lockhart, 1972; Popham, 1969; Salisbury, Richards, & Klein, 1985). In addition, the experimental group also received specific feedback on their performance on the exercises and not just the answers subsequently given to both groups.

The strong effect for Module 4 is most likely a function of the nature of the content of that module. The skills learned in Modules 1–3 were prerequisite skills for Module 4, which required much more complex behavior than the preceding modules. In Module 4, students had to use the DOS commands learned independently in Modules 1–3 and integrate them in the proper order with the new commands from Module 4 to construct a batch file. This is a complex task that requires application and synthesizing of new information with prior knowledge.

The increased practice by the experimental group could be expected to have its greatest effect on complex learning tasks such as the one in Module 4. Sullivan and Higgins (1983) stress the importance of more frequent and individual practice for conceptually difficult learning tasks. Research by Craik and Tulving (1975) and Bretzing and Kulhavy (1979) indicates that practice in elaborating and organizing previously learned information yields better performance in recalling the information.

Subjects whose exercises were collected and graded in the experiment scored 13 percentage points higher (65% as opposed to 52%) than those whose exercises were not collected and graded. However, students in the field test the preceding year had averaged 61%, only four percentage points lower than the experimental group and nine points higher than the control group, even though their instruction was like that of the control group. This apparent anomaly led the investigators to question the course instructors and examine student records for possible explanations.

The course instructor reported that the students in the field test class were highly motivated, much more so than the classes in the experiment. He explained that the semester of the field test was the first time the class had been offered, and only subjects who wanted to take it enrolled because it was strictly an elective course at that time. Students who took the course during the year of the experiment were less self-selected because it was known that the course was soon to be made a requirement, which it now is. Examination of students' records revealed that the cumulative undergraduate grade-point average for the field test class was 2.72, as contrasted with 2.53 and 2.57 for the two classes in the experiment.

The present results highlight an important difference of opinion between instructional designers and course instructors that is not unique to this study. Being well versed in the precepts of competency-based instruction, the instructional designer was not satisfied with posttest scores in the 60% range, just as many developers would not be. She had worked very hard to develop the program and, to a large degree, felt personally responsible for student mastery of its content. She wanted higher student achievement and felt that one way to achieve it was to have present and future course instructors collect and grade the homework exercises. She also recommended extending the length of the program to provide more student practice and instruction, a recommendation that was also made by several subjects in the field test and experiment.

The regular course instructor had a different opinion and was refreshingly candid about it. He was satisfied with the level of performance from the field test. During debriefing by the developer, he explained that he considered the learning task to be an important but difficult one, and that the test was very difficult. "The easiest way to get from 60% to 80% or higher," he noted, "is to change the test. Make it easier."

The instructor reported that he was too busy to grade the exercises as specified in the instructor guide, but that he would probably do so if he had more time or a grader for the course. He also was unwilling to allocate more time to the program because of the amount of content to be covered in the rest of the course.

Who's right? Certainly both sides have their advocates. The developer's opinion is consistent with that of instructional designers generally and with current models of competency-based instruction and mastery learning. Instructional developers are trained to attempt to produce consistently high learner performance, and they generally place greater responsibility on themselves, and perhaps on the instructor, for student learning.

The instructor's opinion, on the other hand, reflects an attitude of many university faculty members and is quite common in disciplines such as mathematics and the hard sciences, including engineering. Many faculty do not expect, or in some cases even want, an average performance level above 60%. Compared to instructional developers, such faculty members are not only likely to accept lower levels of student performance, but also to hold the students more responsible for their own learning. University faculty generally are not trained in the idea that they should be responsible for a high level of student achievement. In fact, this idea is likely to be a novel one to many faculty members.

In essence, the difference of opinion between the designer and the course instructor centers on the question, "Who's responsible for student learning?" Instructional designers are more likely to place heavier responsibility on themselves and the instructors using their programs, whereas many instructors, perhaps especially at the university level, may place greater responsibility on the student. The instructor, of course, has the last word about how the program will be used.

Who's responsible for student learning?

Obviously, it is important for instructional designers to recognize the practices and perspectives of the types of instructors who may use the programs that the designers produce. Recommendations and directions in an instructor guide, even when well founded, are not likely to significantly change the teaching behavior of faculty members whose teaching habits and work priorities are well established.

There are no easy answers for instructional designers to this dilemma. Research by Reiser and Sullivan (1977) demonstrates that college undergraduates perform better under instructor monitoring than under their own. Still, considering the present results, it seems important for instructional designers to develop programs, at least for higher education levels, that include the option to place as much responsibility as possible on students for their own learning. The student materials for such programs would emphasize the importance of completing all learning activities and would include the exercises, self-tests, and feedback needed for acquiring the learning content and monitoring one's own performance. This approach would enable the instructor to either personally direct and monitor student performance or to turn much of that responsibility over to the students themselves.

The authors gratefully acknowledge the conscientious reviews and the thoughtful comments and suggestions from Dr. Norman Higgins and the anonymous reviewers.

. . . it is important for instructional designers to recognize the practices and perspectives of the types of instructors who may use the programs that the designers produce.

References

Branson, R. K. (1981). Applications research in instructional systems development. *Journal of Instructional Development, 4*(4), 14–31.

Branson, R. K. (1988). Why the schools can't improve; The upper limit hypothesis. *Journal of Instructional Development, 10*(4), 15–26.

Bretzing, B. B., & Kulhavy, R. W. (1979). Notetaking and depth of processing. *Contemporary Educational Psychology, 4*, 145–153.

Burkman, E. (1987). Factors affecting utilization. In R. M. Gagné (Ed.), *Instructional technology: Foundations*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Campbell, D., & Stanley, J. C. (1963). *Experimental and quasi-experimental designs for research*. Chicago: Rand McNally.

Craik, F. I. M., & Tulving, E. (1975). Depth of processing and the retention of words in episodic memory. *Journal of Experimental Psychology, 104*(3), 268–294.

Craik, F. I. M., & Tulving, E. (1975). Depth of processing and the retention of words in episodic memory. *Journal of Experimental Psychology, 104*(3), 268–294.

Durzo, J. J., & Florini, B. M. (1984). Evaluating instructional development projects: Guidance from the literature on change. In R. K. Bass and C. R. Dills (Eds.), *Instructional development: The state of the art, II*, Dubuque, IA: Kendall Hunt.

Gagné, R. M., & Briggs, L. J. (1979). *Principles of instructional design* (2nd ed.). New York: Holt, Rinehart and Winston.

Hall, G. E., & Loucks, S. F. (1977). A developmental model for determining whether the treatment is actually implemented. *American Educational Research Journal, 14*(3), 263–276.

McCombs, B. L., Back, S. M., & West, A. S. (1984). *Self-paced instruction: Factors critical to implementation in Air Force technical training—A preliminary inquiry*. (TP-84-23). Lowry Air Force Base, CO: Training Systems Division.

Popham, W. J. (1969). Objectives and instruction. In W. J. Popham, E. W. Eisner, H. J. Sullivan, & L. L. Tyler (Eds.), *AERA monograph series on curriculum evaluation*. Chicago: Rand-McNally.

Reiser, R. A., & Sullivan, H. J. (1977). Effects of self-pacing and instructor-pacing in a PSI course. *Journal of Educational Research, 71*(1), 8–12.

Salisbury, D. F., Richards, B. F., & Klein, J. D. (1985). Designing practice: A review of prescriptions and recommendations from instructional design theories. *Journal of Instructional Development, 8*(4), 9–19.

Stake, R. E. (1967). The countenance of educational evaluation. *Teachers College Record, 68*(7), 523–540.

Sullivan, H. J., & Higgins, N. (1983). *Teaching for Competence*. New York: Teachers College Press.

Weber, S. L., & López, C. L. (1987). *DOS literacy: Commands, directories, and batch files*. Tempe, AZ: Arizona State University.