# The Systematic Approach in Teaching Database Applications: Is There Transfer When Solving Realistic Business Problems?

**Catherine Chen**

**Charles Ray**

*To be successful in their careers, business graduates need to be able to use the computer as a tool to solve problems and to make decisions. This study investigated the effectiveness of a commonly used systematic approach to teach database application skills. Qualitative and quantitative analyses revealed that after five weeks of step-by-step instruction, students were not able to transfer their database application skills to support the processes of solving a realistic business problem. Lack of retention and the inability to apply the appropriate procedures to support problem solving were the main difficulties; in turn, students either made poor conclusions or were unable to make any conclusion. We identify implications for technology educators and authors of learning materials.*

## Introduction

The use of computer technology in the work force has increased rapidly in the past two decades. To meet the demand for computer skills in the work force, business educators have included computer applications as part of the business education curriculum. Ultimately, business students need to have these computer skills to be productive and to solve business problems on their jobs.

There are two major concerns in teaching computer skills: what to teach and how to teach effectively. Studies are conducted regularly to investigate what to teach. Researchers use various investigative techniques to gather data on the computer skills needed in various positions at companies of various sizes, geographic locations, and types of businesses (Arney, 1998; Frueling, Kerin, & Sebastian, 1997; Kim, Keith, & Perreault, 1995; North & Worth, 1997; Perry, 1998; Zhao & Alexander, 2002). Few studies, however, have been conducted to investigate how to teach computer skills effectively. After reviewing all training research published in the top 12 management information systems journals from 1987 to 1996, Gjestland, Van Slyke, Collins, and Cheney (1997) found only five articles on the effectiveness of training methods, and none reported on database training. Three of the studies were conducted with university students, and two were with business employees. Nonetheless, the majority of computer application instructors use the systematic teaching approach (Lambrecht, 1999; McEwen, 1996), and many business teachers consider this approach to be the most effective for teaching software skills (McEwen, 1996). In the systematic teaching approach, students are closely guided in learning to use software, but no empirical evidence is available on the ability of systematic instruction to prepare students to transfer their computer skills to solve business problems.

This research investigates students' ability to solve a business problem individually or within groups after five weeks of step-by-step instruction on the database application Microsoft Access. The instructor used a textbook that provides a scenario of a business situation at the beginning of each chapter and leads students through solving the

*Catherine Chen is Associate Professor, Information Systems and Operations Management Department, Ball State University, Muncie, Indiana.*

*Charles Ray is Professor Emeritus, Information Systems and Operations Management Department, Ball State University, Muncie, Indiana.*

business problem using various Access features. At the end of each chapter, the textbook provides cases with simulated business problems, including directions to solve each case problem. Although some tasks needed to solve the problem require study beyond the textbook, the text usually suggests the tasks to be performed and the expected result. This format is common in systematic, step-by-step instruction.

## Purpose

The purpose of this research was to investigate students' ability to apply their database skills to solve realistic business problems as either individuals or within groups after receiving step-by-step instructions. Specifically, this study investigated the following questions:

1. Were students able to solve realistic business problems using a database software application?
    a. How well did students use the database software to support their problem-solving processes?
    b. What was the quality of the students' decisions?
2. What difficulties emerged as students attempted to solve a problem?
3. Were there any differences in students' problem-solving ability as individuals or within groups when using database software?

## Review of Literature

The review of literature discusses the conceptual framework for the study: the use of computers as cognitive tools, the systematic teaching approach, and the issue of transfer.

### *Computers as Cognitive Tools*

Throughout human history, the use of tools that assist in cognitive processing has been documented. "Environments in which humans live are thick with invented artifacts that are in constant use for structuring activity, for saving mental work, or for avoiding error" (Pea, 1993, p. 48). Notepads, pens, and calculators, and more recently, computers are tools that help us process information (Salomon, Perkins, & Globerson, 1991). Pea (1993) referred to the use of artifacts and tools as "distributed intelligence." Using artifacts as cognitive tools helps human beings reach beyond the limitations of the mind in activities like thinking, learning, and problem solving.

As a cognitive tool, computers can be used to perform mechanical operations, such as tedious recalculations or sequencing, and permit higher-order thinking, such as problem formulation and problem analysis (Pea, 1986; Perkins, 1985; Schoenfeld, 1988). It is in this way that computers are used in business and industry as tools for problem solving to increase employees' productivity. Perkins (1993), however, raised the question of the "fingertip effect" (p. 95): Will people take advantage of using the technology simply because it is made available? For users to employ the technology effectively, training on which functionality can be used to solve what types of problems must be provided. Research has shown that students do not use the powerful features of word processors that facilitate the revision and restructuring of text to improve their writing. Instead, they make only minor stylistic and spelling corrections to obtain documents with appealing formats and to reduce typographical errors (Cochran-Smith, 1991; Daiute, 1986).

Without explicit instructions, students do not master the art of using available tools to solve problems (Perkins, 1993); therefore, teachers need to question whether their teaching practices provide explicit instruction to develop students' problem solving skills. When applying Perkins' theory in teaching software applications for solving business problems, teachers should provide explicit instruction on problem-solving steps and how the software can be used to gather and organize data to provide relevant information in various steps of the problem-solving process; in other words, the focus of the instruction should be on using the software as a tool to organize and provide information to assist in problem solving. Providing explicit instruction only on the steps of using a particular software feature, such as creating a report, is not sufficient to teach students to use the software as a tool for problem solving.

## Systematic Approach to Teaching and Learning

Psychological theories of learning are the basis of curricular and instructional design, and the assumptions of behaviorism provide strong support for the systematic learning approach (Fosnot, 1996). In this view, learning is regarded as a behavioral response to physical stimuli. It assumes that "(1) ... observations, listening to explanations from teachers who communicate clearly, or engaging in experiences, activities, or practice sessions with feedback will result in learning and (2) ... proficient skills will quantify [accumulate] to produce the whole, or more encompassing concept" (p. 9). With these basic assumptions, teachers who use the systematic approach in teaching computer applications use textbooks that provide step-by-step instructions to guide students through exercises. The textbooks' colorful screen illustrations merely present a reference point for students to assess their progress. The use of the application is decomposed into subskills, and these subskills are sequentially presented to students.

McEwen (1996) surveyed 250 National Business Education Association members and found that the most frequently used strategy in teaching software skills was a "teacher-centered instruction method that usually involves step-by-step directions to complete tasks" (p. 17). Lambrecht (1999) conducted research with exemplary secondary and postsecondary office technology programs to investigate teaching and learning practices used in college. The participating programs were selected for the study by having consistently high employment placement rates, being responsive to new technology and current employment needs, and initiating program innovations. The research revealed that "virtually all of the programs provided introductory and intermediate-level computer training using systematically oriented instructional approaches" (p. 53). In this approach, the teacher may or may not demonstrate the use of certain application features, and then students follow the step-by-step instructions in a textbook to complete projects. This approach almost guarantees that students can produce perfect results. Although the textbook may present the steps in the context of various business scenarios, students are not asked either to analyze the problem or to articulate the procedures themselves to solve the problem. Homework instructions may not be as detailed, but students are usually told what to do to get the correct answers and solutions. Since students are guided so closely, their work shows no indication of how well the students can apply what they learn in different situations, particularly on the job.

Students' ability to transfer what they learned in school to their jobs has been a concern in education (Hiebert et al., 1996). In his presidential address at the 1999 American Educational Research Association Annual Meeting, Schoenfeld (1999) identified the issue of transfer as one of the six challenges in theoretical development with profound significance for practice in the 21st century. If the goal is to prepare students to use computer skills to be productive and solve problems on the job, teachers need to question the ability of step-by-step instruction to prepare students to transfer and apply the computer skills they learned to solve realistic business problems. This teaching approach does not provide the opportunity for students to learn the process of identifying information for problem solving.

In addition, the step-by-step approach is typically an individualized approach to teaching. Students work individually with the textbook and complete projects and assignments at their own pace. Yet, businesses are looking for employees who can work in groups and be team players (McLaughlin, 1995; Secretary's Commission on Achieving Necessary Skills [SCAN], 1991). Organizations are increasingly using groups to improve quality, speed, innovation, and customer satisfaction (Andrews, 1995). The step-by-step approach does not provide students the opportunity to work within groups, however, and little is known about how students perform within groups after receiving step-by-step, individualized instruction.

## The Issue of Transfer

The main purpose of instruction is to support the transfer of learning from classroom situations to new situations (Resnick, 1989). The issue of transfer is of particular importance if the instruction is to prepare students for employment (Lambrecht, 1999; Sternberg & Spear-Swerling, 1996). Students learn how to sequence, extract, and organize data into information in a database application class, for example, by creating filters, queries, forms, and reports. When employed in business, these students will be expected to use the application to organize business information to solve problems and to make decisions.

Studies in cognitive psychology have found that students had difficulty transferring and using previously learned knowledge to solve novel problems (Frederiksen, 1984; Saloman & Perkins, 1987; VanderStoep & Seifert, 1994). When students transferred what they had learned, they transferred the knowledge only when the tasks shared common elements (Anderson, 1990; Resnick, 1989). When students were exposed to multiple examples before solving a new problem, however, transfer rates were higher. Transfers were also more likely if students were asked to compare solution-relevant features in examples (VanderStoep & Seifert, 1994).

Using problem selection as an example, Perkins (1993) pointed out that sometimes the lack of transfer was a misdiagnosis. Conventional education selects problems for students but expects students to be able to identify everyday life problems and solve them. Students frequently fail to see opportunities to transfer what they have learned, because the students never had the chance to learn the problem selection process. Applying Perkins's observation to teaching computer applications with a systematic approach, the question arises as to whether students have had the opportunity to learn the process of organizing data for problem solving and decision making, when the teaching approach has already provided all the steps needed to solve the problems successfully.

## Methods

We collected multiple sets of data: (a) a demographic questionnaire to assess the subjects' background information, (b) audiotapes and video tapes of the subjects engaged in a problem-solving activity, (c) a floppy diskette with files created by the subjects, and (d) the conclusions made and written by the subjects at the end of the problem-solving activity.

### *Data Familiarity*

To familiarize students with the database structure and the nature of the database tables used in the problem-solving activity, we devoted one class period to introduce students to the database file. We issued each student a floppy diskette with a database file that contained the same data table structure as that used in the study; however, the study used a different database file with different content in some tables. Each student worked on the computer to examine the database tables while we used a screen projector to explain the content of each data table. We verbally guided students through the process of solving a sample business problem by using the database file to gather relevant information. With our assistance, students worked individually to solve the problem and then turned in a written statement of their conclusions. We answered students' questions while they solved the problem.

### *The Problem-Solving Activity*

A realistic business database file was created using the Business Data Generator (Chen, 2000). Business Data Generator is software that generates database files with a large volume of realistic business data for instructional use. The database file contained nine tables with the number of records ranging from 6 to 827. Unlike most textbook problems, the problem used in this study did not provide the student the tasks needed to solve the problem. The problem stated that the sales manager noticed that the performance of one of her six sales representatives seemed low recently. She wanted information on the poor performer's performance compared with those of other sales representatives and wanted to know if she should be concerned. Participants needed to analyze the problem, decide the measurement of good performance (i.e., high sales amount, frequent customer contacts, high number of transactions generated, high profits made, etc.), use appropriate Access procedures to gather the information, and make a conclusion based on the information they gathered. Although students had used a similar database to solve a class problem and the majority of them (18 of 26 in groups and 9 of 10 individuals) were business students, the problem was selected for the study because it did not require students to possess in-depth business knowledge in a particular area to solve it successfully. The focus of the problem-solving activity was on the students' ability to use the software to help them gather relevant information.

The problem-solving activity was conducted within two weeks of the conclusion of database instruction. Subjects were randomly assigned to perform either as individuals or in teams, and all subjects were allowed to sign up for office hours provided by the two researchers. Each individual or team had one hour to complete the problem-solving activity. All activities were conducted in the same office with the same setting: a computer with Pentium processor, an inkjet printer, a tape recorder, and a video recorder. The textbook was available for their reference on request.

At the beginning of each problem-solving activity each subject completed a demographic questionnaire. Subjects were then given a printed description of the business problem and a paper model of the database structure for their reference. They were told that they would have one hour to solve the problem to their own satisfaction and that they would be required to verbalize their thoughts as they worked. They were also told that they had to solve the problem on their own, and the researcher was not able to answer any questions. Both the tape recorder and the video recorder were activated when the subjects started the problem-solving activity. The researchers observed the subjects during the activity and did not speak except to remind the subjects to keep verbalizing their thoughts. When the subjects indicated that the problem was solved or that they did not have the desire to continue, the recordings were stopped.

### *Subjects*

This study used protocol analysis research procedures. Since protocol analysis procedures are complex and time-consuming, these studies usually involve a small number of subjects (cf., Gall, Borg, & Gall, 1996).

For this study, we recruited subjects from 44 students enrolled in two sections of an advanced microcomputer applications for business course at a Midwestern university. The same instructor taught the two sections of the computer course using the same textbook and materials. To obtain an equal number of problem-solving activities in the individual category and team category, we randomly assigned 11 students to solve the problem as individuals and 33 students to 11 teams of 3 students. Some students withdrew from the class, and others did not attend at the time of the problem-solving activity. We collected data from 11 students who solved the problem as individuals and from 8 teams of 3 students and 1 team of 2 students for a total of 37 students. To eliminate students' concerns about how their problem-solving performance might affect their course grades, we conducted the problem-solving activity after completing the Access unit and administering the Access examinations. Students knew that they would receive a fixed number of bonus points regardless of their performance on the problem-solving activity.

### *Subject Demographics*

The demographic questionnaire asked for gender, college major, and year in school. We generated cross-tabulation tables to compare the students who solved the problem individually with those in teams in relation to each demographic variable, and used Pearson chi-square tests to determine if students differed in terms of demographic characteristics. These tests showed that the students working in groups were similar to those working individually in gender, years in school, and major field of study, as none of the chi square values were statistically significant.

*Prior database work experience and prior database courses*. The demographic questionnaire also asked students about their prior database-related work experience and prior database courses taken. Again, the Pearson chi-square values were non-significant. Students who solved the problem individually and who solved the problem in teams had similar experience in using database software on the job and were not significantly different in terms of database courses taken prior to this study.

### *Protocol Analysis Procedures*

A qualitative research approach can generate data resulting in patterns of events (Kincheloe, 1991), and can help investigators identify students' ability to use software applications to gather relevant information about their problem solving. Ericsson and Hastie (1994) maintain that verbal protocols offer a rich description of the processes of each student and make it possible for researchers to compare the processes of different students who are solving the same problem. We used protocol analysis research procedures to investigate how students solve problems on their own, to determine the difficulties that students encounter when solving problems, and to identify the steps students use to determine the answers. We were thus able to gather detailed information on students' ability to use the database application to support their problem-solving processes.

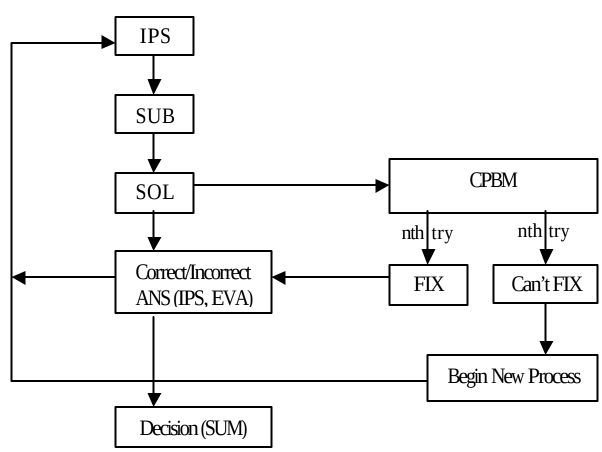
We analyzed the verbal records or protocols of the problem-solving activity using protocol analysis procedures developed by Ericsson and Simon (1993). We transcribed, segmented, and coded the verbal protocols. We coded segments consisting of discrete thoughts or phrases according to the categories presented in the Computer-Aided Problem-Solving Model (Chen, 1999). The model combines a problem-solving control structure for analyzing problem-solving skills in the social sciences with theories of human-computer interaction. Chen developed the model for researchers to plot subjects' verbal protocols collected when using computers to solve business problems. Figure 1 illustrates the model.

For easy reference, we assigned codes to the protocols. Protocols generated by individual subjects were lettered from A to K, and protocols generated by teams were numbered from 1 to 9.

One method of analyzing qualitative data is to look for patterns in data so that general statements about the phenomenon can be made (Potter, 1996). To use this inductive method to analyze the data, we generated variables for analyses from the protocols themselves. We examined the protocols, videotapes, and files to identify common themes, the subjects' ability to use the Access software to support problem solving, and the quality of the subjects' conclusions. Planning emerged as a common theme. We constructed a rating system based on the planning demonstrated by the subjects: 1—no planning; 2—some planning, not executed/unable to execute; and 3—detailed planning, able to achieve what was planned. We also constructed a rating system based on the quality of the conclusions stated by the subjects: 0—no conclusion, 1—inaccurate or misinterpreted results, 2—somewhat accurate but misinterpreted results, 3—somewhat logical conclusion based on limited support, 4—good conclusion but not well supported, 5—good conclusion and well supported.

Based on the procedures used in the problem-solving activities by the subjects, we identified 14 variables: successful queries, successful relationships, successful reports, successful forms, unnecessary queries, queries with unnecessary

**Figure 1: Computer Aided Problem Solving Analysis Model**




Key: IPS—Interpret problem statement, data, or answers
SUB—State subproblem
SOL—State solution
EVA—Evaluate solution or answers
ANS—State computer's response
FIX—Solve the computer problem
SUM—Summarize the decision
CPBM—State computer problem

tables, queries with undetected incorrect results, unnecessary relationships, unnecessary reports or reports with incorrect results, unnecessary forms, unsuccessful queries, unsuccessful relationships, using help, and “being stuck.” We categorized subjects as “stuck” if the problem-solving process was aborted because they did not know how to proceed, could not figure out how to perform a procedure, or could not fix their errors.

Together with the ratings on planning and the quality of conclusions, we recorded and analyzed the subjects’ use of the above 14 procedures, using the Mann-Whitney *U* test to compare individual and team problem solving. The Mann-Whitney *U* test is one of the most powerful nonparametric tests that can be used to test the difference between the means of two independent samples. It is a useful alternative to the *t* test when samples are small and of unequal sizes (Howell, 1987; Minium, 1978).

## Findings

We examine the collected data both qualitatively and quantitatively to address the research questions.

### Research Question 1

Research question 1 investigated the students’ ability to solve a realistic business problem using a database software application. To provide answers to this research question, we examined the collected data to answer two sub-questions: (a) How well did students use the database to support their problem-solving processes? and (b) What was the quality of the students’ decisions?

Table 1 lists the planning and the quality of conclusions and various procedures used by individual subjects, and Table 2 lists the planning, quality of conclusions, and procedures used by teams. For example, the Query row under the ‘Successful procedures’ sub-heading indicates the number of queries that individuals or team members created during problem-solving activities. The rows under the ‘Successful but unnecessary procedures’ indicate the number of unnecessary queries, relationships, reports, or forms that participants created in the process of solving the problem. These numbers illustrate the participants’ ability to use the software, but they also illustrate their inability to judge the irrelevance of these procedures in their problem-solving processes.

*Database skills.* As stated in the Protocol Analysis Procedures, planning emerged as a common theme. Table 1 and Table 2 show that 6 of 11 individuals and 5 of 9 teams performed no planning. Only two individuals and two teams were able to plan and achieve what was planned. Most subjects in this study immediately opened the database file and attempted Access procedures without determining what they intended to accomplish. The only two individuals who were able to solve the problem were those who planned well prior to using the Access features. Only one of the two teams who were able to solve the problem had a good plan that was well executed.

Most subjects, in addition, did not examine the paper copy of the database structure to understand the data available to help them solve the problem. Both individuals and teams opened the database file immediately without planning and wasted time in creating, or attempting to create, a great deal of unnecessary queries, relationships, reports, and forms. For example, subject A created three queries successfully, but none of them helped solved the problem because they did not provide relevant information. Subject A wasted time creating queries that contained unnecessary tables, and furthermore, subject A did not detect that the results provided by these queries were incorrect results.

The majority of the queries created by subjects A, B, C, D, F and teams #6 and #9 were unnecessary queries that did not provide useful information. It was common for subjects to include unnecessary tables in their queries as well. An alarming number of subjects (six individuals and four teams) did not detect errors in the query results. They assumed that once a query generated results without problems, the results were correct. Most subjects did not perform cross-reference checks. Another theme that emerged from the protocols was that subjects appeared to be “fishing” for procedures. They went through all the options in the pull-down menus one by one looking for procedures that might help.

As indicated in Table 1 and Table 2, 9 of 11 individuals and 7 of 9 teams were not able to continue solving the problem because they were "stuck." When they were stuck, some subjects made conclusions based on the information that they were able to gather up to that point, but others could not come to a conclusion. Although a majority of the subjects used a great deal of unnecessary procedures and were not able to solve the problem to their satisfaction, none of the individuals and only one team attempted to use the help feature to refresh their knowledge of the software.

*The quality of students' decisions.* As indicated in Tables 1 and 2, only one team and two individuals were able to make good conclusions supported by their use of the database application. Four individuals and two teams were so "stuck" that they did not make conclusions. For the rest of the subjects, the conclusions that they made were "somewhat logical" at best.

### *Research Question 2*

Research question 2 investigated the difficulties students encountered when solving a realistic business problem. Verbal protocols were examined to reveal students' difficulties. The most common difficulties were that either the subjects did not remember how to carry out a procedure, or they did not know which procedure to use. Although the

**Table 1: Individuals' Planning, Quality of Conclusions, and Number of Procedures Used**

| | Individuals | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | B | C | D | E | F | G | H | I | J | K |
| Planning (Rating: 1, 2, or 3) | 1 | 1 | 2 | 1 | 2 | 1 | 3 | 2 | 3 | 1 | 1 |
| Conclusion (Rating: 0, 1, 2, 3, 4, or 5) | 1 | 2 | 0 | 0 | 2 | 0 | 5 | 1 | 5 | 3 | 0 |
| Stuck | Y | Y | Y | Y | Y | Y | N | Y | N | Y | Y |
| Using Help | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Successful procedures (may not generate useful info) | | | | | | | | | | | |
| Query | 3 | 3 | 4 | 2 | 2 | 1 | 5 | 2 | 4 | 1 | 1 |
| Relationship | 0 | 1 | 2 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 1 |
| Report | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 2 |
| Form | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
| Successful but unnecessary procedures | | | | | | | | | | | |
| Query | | | | | | | | | | | |
| Unnecessary queries/Queries without useful info | 3 | 2 | 3 | 2 | 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| Queries with unnecessary tables | 3 | 3 | 0 | 1 | 1 | 1 | 2 | 1 | 0 | 0 | 0 |
| Queries with undetected incorrect results | 3 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| Relationship: Unnecessary relationships | 0 | 0 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Report: Unnecessary reports/Incorrect reports | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 2 |
| Form: Unnecessary forms | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
| Unsuccessful procedures | | | | | | | | | | | |
| Query | 1 | 1 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Relationships | 0 | 0 | 1 | 1 | 2 | 0 | 0 | 0 | 0 | 2 | 0 |
| Total of wasteful procedures or steps | 11 | 7 | 11 | 7 | 6 | 4 | 3 | 4 | 1 | 3 | 6 |

**Notes:** Planning: 1—no planning, 2—some planning, not executed/unable to execute, 3—detailed planning, able to achieve what was planned.
Conclusions: 0—no conclusion, 1—inaccurate or misinterpreted results, 2—somewhat accurate but misinterpreted results, 3—somewhat logical conclusion based on limited support, 4—good conclusion but not well supported, 5—good conclusion and well supported.

**Table 2: Teams' Planning, Quality of Conclusions, and Number of Procedures Used**

| | Teams | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | #1 | #2 | #3 | #4 | #5 | #6 | #7 | #8 | #9 |
| Planning (Rating: 1, 2, or 3) | 1 | 1 | 2 | 1 | 1 | 3 | 2 | 3 | 1 |
| Conclusion (Rating: 0, 1, 2, 3, 4, or 5) | 0 | 1 | 3 | 3 | 2 | 0 | 3 | 5 | 3 |
| Stuck | Y | Y | N | Y | Y | Y | Y | N | Y |
| Using Help | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Successful procedures (may not generate useful info) | | | | | | | | | |
| Query | 1 | 2 | 8 | 1 | 6 | 1 | 2 | 6 | 1 |
| Relationship | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Report | 1 | 0 | 0 | 1 | 1 | 2 | 0 | 0 | 2 |
| Form | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Successful but unnecessary procedures | | | | | | | | | |
| Query | | | | | | | | | |
| Unnecessary queries/Queries without useful info | 0 | 1 | 0 | 0 | 6 | 0 | 0 | 0 | 1 |
| Queries with unnecessary tables | 0 | 0 | 0 | 0 | 0 | 1 | 2 | 0 | 1 |
| Queries with undetected incorrect results | 0 | 0 | 0 | 0 | 6 | 1 | 2 | 1 | 0 |
| Relationship: Unnecessary relationships | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Report: Unnecessary reports/incorrect reports | 1 | 0 | 0 | 1 | 1 | 2 | 0 | 0 | 2 |
| Form: Unnecessary forms | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Unsuccessful procedures | | | | | | | | | |
| Queries | 0 | 3 | 0 | 1 | 0 | 0 | 2 | 0 | 2 |
| Relationships | 0 | 4 | 0 | 0 | 0 | 0 | 3 | 0 | 0 |
| Total of wasteful procedures or steps | 1 | 8 | 0 | 2 | 14 | 4 | 9 | 1 | 7 |

**Notes:** Planning: 1—no planning, 2—some planning, not executed/unable to execute, 3—detailed planning, able to achieve what was planned.
Conclusions: 0—no conclusion, 1—inaccurate or misinterpreted results, 2—somewhat accurate but misinterpreted results, 3—somewhat logical conclusion based on limited support, 4—good conclusion but not well supported, 5—good conclusion and well supported.

problem-solving activities were conducted within two weeks of completing the Access unit, retention was a major problem. The phrase "I just don't remember" frequently appeared in the transcripts.

Subjects also indicated not knowing what to do. They stated that it was easy to follow the instructions in the book or the instructor's demonstration, but when they needed to figure out the problem on their own, they had a great deal of difficulty. As revealed in Research question 1, the majority of the subjects were not able to solve the problem to their own satisfaction.

To further examine the effectiveness of the systematic teaching approach, the numbers of relevant procedures that the subjects used in the five-week instruction were recorded. The subjects had created 25 queries, created 10 reports, and used the help feature 5 times in the course of the five-week unit as instructed by the textbook. Both individuals and teams, however, had difficulty either remembering how to perform the procedures or knowing when to use a procedure during the problem-solving exercise. They could follow directions in the book, but they were not able to select the appropriate procedures when solving a problem without specific instruction. They also had used the help feature to resolve computer problems while taking the class, but few subjects attempted to use help for assistance during this study.

### *Research Question 3*

The last research question addressed the differences in students' problem solving as individuals or within groups when using database software. To answer this research question, a quantitative approach was used. The Mann-Whitney *U* test was applied to the data listed in Table 1 and Table 2. Table 3 shows the results.

As indicated in Table 3, there were no significant differences in individual and team problem solving using a database application. Both individuals and teams planned poorly or did no planning, had difficulty either remembering how to use or identify the relevant Access procedures to gather information, failed to use help, and made either poor conclusions or no conclusions.

**Table 3: Comparison of Individual and Team Problem Solving**

| Variable | P value (2-tailed) |
|---|---|
| Planning rating | .966 |
| Conclusion rating | .413 |
| Stuck on problem | .827 |
| Using Help | .269 |
| Query | |
| Successful queries | .906 |
| Unnecessary queries | .115 |
| Unnecessary tables in queries | .161 |
| Queries with undetected incorrect results | .710 |
| Unsuccessful queries | .737 |
| Relationship | |
| Successful relationships | .177 |
| Unnecessary relationships | .625 |
| Unsuccessful relationships | .778 |
| Report | |
| Successful reports | .425 |
| Unnecessary/incorrect reports | .352 |
| Successful by unnecessary forms | .191 |

**Note:** P value is the probability of Type I error. P values should be less than .05 or .01 to be considered statistically significant.

## Conclusions

As mentioned in the Introduction, the most frequently used strategy in teaching software skills involves step-by-step directions to complete tasks, and the majority of business teachers who responded to McEwen's 1996 survey considered their approach the most effective for teaching these skills. No empirical evidence was available, however, on the ability of systematic instruction to prepare students to transfer their computer skills to solve business problems. The conclusions of this study provide information on students' ability to solve business problems after receiving only step-by-step instructions. The data were collected from a four-year university with traditional business students; therefore, the results can be generalized only to this student population.

### *Problem-Solving Ability*

Students who learn Access by systematic, tutorial-type instruction alone were not able to solve business problems using a database as the source of information. Since the materials used in this approach emphasize software techniques and procedures, students demonstrated poor problem-solving skills. The students attempted to solve problems without defining the problem, planning their problem-solving activities, or identifying sets of data that would help them solve the problem. Students who participated in this study did not take the time to examine how data stored in a database could provide them with relevant information and help them solve the problem. They tended to open the database file immediately without any examination of the structure of the data and started "fishing" through menu options in the application for procedures that might give them some information. As a result, the quality of their decisions was poor. After exposure to numerous demonstrations and exercises involving database tasks such as creating queries, creating reports, and using online help facilities, students were not able to use these procedures to solve a business problem.

### *Difficulties Encountered*

When asked to use the database application to solve a business problem after completing an Access unit, the students' recollection of the database procedures needed to produce results related to the problem was insufficient or nonexistent. They did not recall how relationships in a relational database could be established to enable them to pull valuable information from related tables. When solving the business problem, they experienced considerable difficulty, both with the choice of database tool (query, report, etc.) and with the correct procedure for effectively using those tools.

Students experienced difficulty when they attempted to transfer what they learned in systematically oriented database software instruction to a business problem-solving situation. This conclusion supports other research reported in the literature (Frederiksen, 1984; Saloman & Perkins, 1987; VanderStoep & Seifert, 1994). Perkins (1993) asserted that students have difficulty transferring problem selection skills from classrooms to everyday life if they were taught procedures for solving pre-identified problems without sufficiently emphasizing problem solving techniques. This study provides conclusions that are similar to Perkins' assertion in that students have difficulty using database software to solve business problems if they are taught the steps to use the software features without sufficient instructions on how to use these features to gather relevant information in the problem-solving processes.

### *Individual Versus Team Performance*

Based on the results of this study, students who received a systematic teaching approach did not perform better when working in teams than when working as individuals. Although students who solved problems in teams had the opportunity to consult with one another, they did not plan better, did not retain what they learned better, and did not make better conclusions when solving business problems.

## Implications

The findings of this study carry implications for information technology educators, corporate trainers, and authors of learning materials. To make sure that learners are able to plan for problem-solving activities using database information, directions and techniques in problem solving should be incorporated into instruction. Preliminary planning for problem solving tasks, a process bypassed by most subjects in this study, should be emphasized. This planning includes identifying the core problem-solving task, selecting sets of data that could impact the solution, and evaluating the relationship of extracted information to the identified task.

While systematic, step-by-step tutorials are commonly used to familiarize students with database procedures and processes, additional instruction appears necessary for retention. So that learners comprehend the value of database software as a tool for decision making and problem solving, activities requiring the use of the software for solving unstructured problems should be incorporated. Systematic, step-by-step instruction works well for learning fundamentals. After initial instruction with the basics, the teaching process should progress to applications that demonstrate how to determine the course of action necessary to extract problem-related data from the database, not merely how to follow a prescribed course of action.

Finally, educators who develop and deliver database instructional materials should strengthen components that articulate the power and necessity of relationships in relational database software. Since complex, realistic business problems often require the problem solver to gather information from multiple sources or tables, as in this study, learners should receive extensive practice that helps them view the three-dimensional nature of databases. Such exposure could help them comprehend the value of converting two-dimensional data collections into valuable information. For example, students should understand that reports could be used to make calculations, organize information, and summarize information related to a particular problem.

Moving away from systematic instruction to an unstructured activity after presenting basics should strengthen learners' understanding of database capabilities and procedures. By allowing students to decide which database procedures to use to organize data into relevant information, the unstructured activity provides an opportunity for students to learn the process of using computers to extract valuable information for problem solving and decision making. Additional research should focus on a combination of software instruction and problem-solving instruction to measure the effect of this combination on the learner's ability to retain knowledge of procedures and to use the software as a tool to extract information to solve problems. Additional experimental research is needed to investigate the effectiveness of various techniques of teaching students to use database applications to solve business problems. Since students typically have more difficulty with database applications than with word processors or spreadsheets, replicating studies using other software applications will provide information on the effectiveness of the systematic approach of teaching students to apply software skills on the job.

## References


Anderson, J. R. (1990). *Cognitive psychology and its implications* (3rd ed.). New York: W. H. Freeman and Company.

Andrews, K. Z. (1995). Cross-functional teams. *Harvard Business Review, 73*(6), 12-13.

Arney, J. B. (1998). A comparison of computer skills needed by business administration graduates as perceived by colleges and universities and human resource managers in Indiana. *NABTE Review, 25*, 44-48.

Cochran-Smith, M. (1991). Word processing and writing in elementary classrooms: A critical review of related literature. *Review of Educational Research, 61*, 107-155.

Chen, C. S. (2000). Using realistic business data in teaching business problem solving. *Information Technology, Learning, and Performance Journal, 18*(2), 41-50.

Chen, C. S. (1999). A protocol analysis model for investigating computer supported problem-solving activities. *Information Technology, Learning, and Performance Journal, 17*(2), 35-43.

Daiute, C. (1986). Physical and cognitive factors in revision: Insights from studying with computers. *Research in the Teaching of English, 20*, 141-159.

Ericsson, K. A., & Simon, H. A. (1993). *Protocol analysis: Verbal report as data* (Revised ed.). Cambridge, MA: The MIT Press.

Ericsson, K. A., & Hastie, R. (1994). Contemporary approaches to the study of thinking and problem solving. In R. J. Sternberg (Ed.), *Thinking and problem solving* (pp. 37-79). New York: Academic Press.

Fosnot, C. T. (1996). Constructivism: A psychological theory of learning. In C. T. Fosnot (Ed.), *Constructivism: Theory, perspectives, and practice* (pp. 8-33). Teachers College, Columbia University.

Frederiksen, N. (1984). Implications of cognitive theory for instruction in problem solving. *Review of Educational Research, 54*(3), 363-407.

Frueling, R. T., Kerin, J. D., Sebastian, D. (1997). The information technology age and its impact on business and business education curriculums. *The Delta Pi Epsilon Journal, 39*(2), 64-75.

Gall, M. D., Borg, W. R., & Gall, J. P. (1996). Educational research: An introduction (6th ed.). White Plains, NY: Longman Publishers USA.

Gjestland, C., Van Slyke, C., Collins, R., & Cheney, P. (1997). *End user training: A decade of research*, (ERIC Document Reproduction Service No. ED422953).

Hiebert, J. Carpenter, T. P., Fennema, E., Fuson, K., Human, P., Murray, H., Olivier, A. & Wearen, D. (1996). Problem solving as a basis for reform in curriculum and instruction: The case of mathematics. *Educational Researcher, 25*(4), 12-21.

Howell, D. C. (1987). *Statistical methods for psychology* (2nd ed.). Boston, MA: PWS Publishers.

Kim, C., Keith, N., & Perreault, H. (1995). A managerial perspective on the importance of information technology topics. *Office Systems Research Journal, 13*(2), 9-14.

Kincheloe, J. L. (1991). *Teachers as researchers: Qualitative inquiry as a path to empowerment*. New York: The Falmer Press.

Lambrecht, J. (October, 1999). *Developing employment-related office technology skills*. MDS-1199. Berkeley, CA: National Center for Research in Vocational Education.

McEwen, B. (1996). Teaching microcomputer software skills. *Business Education Forum, 50*(4), 15-19.

McLaughlin, M. (1995). *Employability skills profile: What are employers looking for?* Office of Education Research and Improvement, (ERIC Document Reproduction Service No. ED399484).

Minium, E. W. (1978). *Statistical reasoning in psychology and education* (2nd ed.). New York: John Wiley & Sons.

North, A. B, & Worth, W. E. (1997). Workplace competencies: Trends in advertised entry-level technology, interpersonal, and basic communication job skills, 1992-1995. *Office Systems Research Journal, 15*(1), 1-6.

Pea, R. D. (1986). *Beyond amplification: Using the computer to reorganize mental functioning* (Technical Report No. 38). New York: Bank Street College of Education, Center for Children and Technology. (ERIC Document Reproduction Service No. ED297706).

Pea, R. D. (1993). Practices of distributed intelligence and designs for education. In G. Salomon (Ed.). *Distributed cognitions: Psychological and educational considerations* (pp. 47-87). New York, NY: Cambridge University Press.

Perkins, D. N. (1985). The fingertip effect: How information-processing technology shapes thinking. *Educational Researcher, 14*(7), 11-17.

Perkins, D. N. (1993). Person-plus: A distributed view of thinking and learning. In G. Salomon (Ed.), *Distributed cognitions: Psychological and educational considerations* (pp. 88-110). New York, NY: Cambridge University Press.

Perry, W. (1998). What software skills do employers want their employees to possess. *Business Education Forum, 50*(3), 35-38.

Potter, W. J. (1996). *An analysis of thinking and research about qualitative methods*. Mahwah, NJ: Lawrence Erlbaum Associates.

Resnick, L. B. (1989). Introduction. In L. B. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 1-24). Hillsdale, NJ: Lawrence Erlbaum Associates.

Salomon, G. S., & Perkins, D. N. (1987). Transfer of cognitive skills from programming: When and how? *Journal of Educational Computing Research, 3*(2), 149-169.

Salomon, G., Perkins, D. N., & Globerson, T. (1991). Partners in cognition: Extending human intelligence with intelligent technologies. *Educational Researcher, 20*(3), 2-9.

Schoenfeld, A. H. (1988). Mathematics, technology, and higher order thinking. In R. S. Nickerson & P. P. Zodhiates (Eds.), *Technology in education: Looking toward 2020* (pp. 67-96). Hillsdale, NJ: Lawrence Erlbaum Associates.

Schoenfeld, A. H. (1999). Looking toward the 21st century: Challenges of educational theory and practice. *Educational Researcher, 28*(7), 4-14.

Secretary's Commission on Achieving Necessary Skills (1991). *What Work Requires of Schools—A SCANS Report for America 2000*. Washington D.C.: U.S. Department of Labor.

Sternberg, R. J., & Spear-Swerling, L. (1996). *Teaching for thinking*. Washington, DC: American Psychological Association.

Van der Stoep, S. W., & Seifert, C. M. (1994). Problem solving, transfer, and thinking. In P. R. Pintrick, D. R. Brown, & C. E. Weinstein (Eds.), *Student motivation, cognition, and learning: Essays in honor of Wilbert J. McKeachie* (pp. 27-49). Hillsdale, NJ: Lawrence Erlbaum Associates.

Zhao, J. J., Alexander, M. W. (2002). Information technology skills recommended for business students by Fortune 5000 executives. *Delta Pi Epsilon Journal, 46*(3), 175-189.