SESSION 9 CUSTOM TOOLS FOR INFORMATION COLLECTION AND MANAGEMENT

Chaired by Howard L. Kaplan, Addiction Research Foundation of Ontario

Subject-pool record keeping: A database application

MARK CARDILLO and DARRELL L. BUTLER
Ball State University, Muncie, Indiana

This paper describes a record-keeping system we developed for our subject pool, utilizing ORACLE database software on a VAX computer. This system consists of three data tables and four processes: set up, updating, report generation, and editing. This article describes the rationale and function of each of these four processes in detail. All input and output is in ASCII format. The combination of ASCII files and the ORACLE database system produces a subject-pool record-keeping system that is adaptable, expandable, and exportable.

Most papers about subject pools have been limited to questions of validity (Burns, 1974; Davis & Fernald, 1975; Jackson, Procidano, & Cohen, 1989; Lipton & Garza, 1978; Miller, 1981; Trice & Bailey, 1986) and ethics (Lindsay & Holden, 1987; Sieber & Saks, 1989). These papers have not been concerned with the administration of subject pools and cannot answer many of the practical questions of interest to experimenters, instructors whose students participate in subject pools, and students. Are people more likely to miss appointments of 30-min experiments or 2-h experiments? What is the distribution best describing weekly participation over a semester? What are the effects of attempts to enhance the educational value of participation? A good record-keeping system is needed to answer these questions. For university subject pools, an organized record-keeping system is also important because it provides useful reports to instructors who provide rewards for students who participate.

Although subject-pool record keeping can be done by hand, we have a sizable subject pool that creates problems

for a hand record-keeping system. Each semester, we have between 1,000 and 3,000 available subjects, who come from about 15 different classes; the average participation of subjects is about 4 h of projects and experiments. The effort required to produce by hand the wide variety of reports we need would be very time-consuming. Furthermore, such a tedious task could have large error rates.

The most attractive method of subject-pool record keeping is the use of a computer system that can minimize errors and provide flexible reports. Information can be stored conveniently in the computer in a database. Data in the database can then be used to create reports. The main drawbacks of utilizing a database system include the high initial software costs (if the software is not readily available), the training and programming experience required to perform nonstandard data manipulations, and the problem that centralized data are more vulnerable to corruption than are noncentralized data (Gorney, 1985, p. 20).

There are two general types of database structures: single tables and multiple lists. Each has its advantages and disadvantages. In a single-table database, all of the information is stored as a single matrix of data, consisting of a series of rows and columns, where each row represents a single record and the columns represent the features of the record. Each row contains the same number of columns, regardless of whether or not descriptive information for that record is stored there. For example, in a subject-pool database, each row could represent a subject and the various columns would contain the subject's name, identification number, class number, and one

This paper was prepared on the basis of the creation of a subject-pool record-keeping system for the Department of Psychological Sciences at Ball State University. We would like to thank Cindy Ruman for her patience and suggestions and Jolanta Czerwinska for helping us hack our way through the denser regions of the computer jungle. The programs (not including ORACLE or ORACLE-related software) and a record-keepers manual are available and will be delivered upon request. Please address correspondence to D. L. Butler, Department of Psychological Science, Ball State University, Muncie, IN 47306 (e-mail: 00dlbutler@bsuvax1.bitnet or 1mdcardillo@bsuvax1.bitnet).

Name	ID #	Class #	Experiments															Total Credits
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
Brand, N.	19803	002	1.0					1.5										2.5
Cole, L.	25894	001				2.0	0.5				1.0						1.0	4.5
Cunningham, J.	12235	003		0.5	0.5			1.5		0.5								3.0
Hays, R.	47303	003									1.0					0.5		1.5
Kaplan, H.	87425	001		0.5			0.5			0.5								1.5
LaCaille, R.	73557	003	1.0						1.5		1.0			1.0				4.5
Merbitz, C.	65277	002	1.0								1.0							2.0
Munneke, D.	66841	003		0.5				1.5										2.0
Sola, C.	32157	001			0.5	2.0			1.5							0.5		4.5
Winterowd, C.	50158	002	1.0			2.0			1.5									4.5

Figure 1. A conceptualization of a single-table database structure.

column for each experiment that was offered during the semester. Figure 1 shows a graphic representation of such a database.

There are advantages and disadvantages to single-table systems. Among the advantages of this type of organization are that (1) the reports are easy to generate because they reflect the structure of the database itself, and (2) navigation through the data is relatively simple and requires little time. However, the disadvantages include the large amount of memory required to hold the table, the difficulty in updating created by the need to specify the column and row of each datum, and the limited range of report generation as a result of the unalterable structure (Gorney, 1985, pp. 12-13; Pepin, 1989, p. 8). In practice, we found another problem with this approach. Each time we chose to modify the system, the resulting program and its database were incompatible with the system that it was replacing. For example, if a new column of information was needed, the programs that utilized that database would have to be modified to take into account the new column. Once the modification was made, the new program could not analyze old data tables and the old programs could not analyze new data tables. This created a variety of problems.

An alternative database format is known as a *relational database management system* (RDBMS). Rather than having a single, centralized data table, an RDBMS utilizes multiple lists of data. Each list is composed of columns and rows. Relationships are "computed" as a user searches for information contained within various lists. A subject-pool record-keeping system based upon multiple lists could have (1) one table that contains subject identification information, such as name, identification number, and class number, (2) another table that contains experiment participation information, such as the participant's identification number and the experiment number they participated in, and (3) a table containing exper-

iment information such as the experiment number and credits with that experiment. Figure 2 illustrates this sort of structure.

There are both advantages and disadvantages to this approach. The advantages of using an RDBMS include the wide variety of reports that can be generated, the ease of updating the information, and the ability to store a maximal amount of information in a minimal amount of space. Furthermore, adding new information as the system evolves does not cause the incompatibility problems to the same extent as with the single-table databases. It is a matter of creating a new table. The only programs to be modified are ones that require the new data.

The disadvantages of using an RDBMS include the increased possibility of logical redundancy within any of the tables (e.g., a student who does the same experiment twice), deficiencies caused by null values (e.g., a report about how many experiments each student has done will not list students who have not participated), and the fact that the more tables required in the generation of a report, the greater the difficulties in the construction of query statements (Rishe, 1988, p. 137-138). Also, there is a tradeoff between report complexity and processing time, although advances in hardware and programming technology have diminished this problem (Pepin, 1989, p. 10). Many of the problems associated with an RDBMS can be anticipated and avoided by following a short list of general search heuristics described by Harter and Peters (1985).

On the basis of a comparison of the advantages and disadvantages of the two kinds of databases, we chose to develop our subject-pool record-keeping system using an RDBMS. This is our third-generation computer record-keeping system. It was designed to provide various kinds of reports for different groups, to be highly flexible and adaptive, to be compatible with a wide variety of hardware configurations, and to be widely supported. After analyzing available software, we chose to base the record-

STUDENT TABLE			PARTICIPATION TABLE		EXPERIMENT TABLE	
Name	ID #		ID #	Exp #	Exp #	Credit Value
Brand, N.	19803	002	47303	14	1	1.0
Cole, L.	25894	001	87425	2	2	0.5
Cunningham, J.	12235	003	25894	9	3	0.5
Hays, R.	47303	003	32157	4	4	2.0
Kaplan, H.	87425	001	19803	1	5	0.5
LaCaille, R.	73557	003	50158	7	6	1.5
Merbitz, C.	65277	002	87425	5	7	1.5
Munneke, D.	66841	003	25894	15	8	0.5
Sola, C.	32157	001	65277	9	9	1.0
Winterowd, C.	50158	002	66841	2	10	1.0
			73557	7	11	0.5
			19803	6	12	1.0
			32157	14	13	2.0
			47303	10	14	0.5
			66841	6	15	1.0
			12235	3		
			12235	8		
			73557	1		
			50158	4		
			12235	6		
			73557	9		
			32157	3		
			65277	1		
			50158	1		
			25894	4		
			32157	7		
			87425	8		
			73557	12		
			25894	5		
			12235	2		

Figure 2. A conceptualization of a multiple-list database structure.

keeping system on ORACLE database management software. ORACLE is one of the most powerful, widely available relational database management systems for use on a wide variety of computer systems at the mainframe level and is increasingly available for use on personal computer systems, such as IBM PCs, PC clones, and Apple Macintosh. Software command scripts are straightforward and easy to create and execute. ORACLE commands can be embedded and executed from within other program files written in other computing languages, such as FORTRAN and C. In addition to the availability of expert advice from staff at many university computer centers, we found that there are a large variety of third-party publications concerning the utilization of ORACLE and SQL*PLUS commands.

THE RECORD-KEEPING SYSTEM

The record-keeping system is summarized in Figure 3. The subject-pool record-keeping system relies on three

main data tables. The first contains the names, section or class numbers, and identification numbers of all students to be included in the subject pool. The second data table contains the experiment identification numbers, along with associated credit values. The third data table contains the students' identification numbers and the number of the experiments in which they participated.

Record keeping involves four processes, as shown in Figure 4. The process of setting up the record-keeping system (e.g., each semester) requires creating a list of students and a list of experiments. Software converts these lists to data tables usable by ORACLE. The second process, updating, consists of adding new records to a participation list and adding new experiments to the experiment list. Software again is used to convert these lists to tables usable by ORACLE. Report generation is the purpose of the system and is the third major process. The fourth process is editing. An editor that provides the ability to edit any record in any list is available. It is useful for maintaining data accuracy.

Figure 3. General overview of the record-keeping system.

Details about the major data tables and processes are provided below.

Set Up

At the beginning of each semester, two ASCII files must be created: STUDENT.TXT and EXPER.TXT. STUDENT.TXT is a list of all students that are available to participate in the experiments offered. This file contains the students' names, the class or section number that each student belongs to, and student identification numbers. An ASCII text file can be created in a variety of ways, including manually entering the information with

a word processor editor. Since our participant pool is quite large, the text file listing is downloaded from the campus' administrative computer.

The experiment list (EXPER.TXT) contains the experiment identification number and the credit value assigned to it. This list is an ASCII text file created using a word processor.

These ASCII files are entered into databases through the use of the ORACLE Data Loader (ODL), the data loader utility program provided with the ORACLE software package. ODL enters the file information into the record-keeping-system data tables in accordance with

Figure 4. The four processes necessary for record keeping.

short control files that provide templates for the formats of the individual ASCII files.

Updating

The record-keeping system can be updated at any convenient time. We do it roughly on a weekly basis. Updating involves files containing information provided about participation in experiments and (if necessary) files containing information about experiment credit.

Participation information is entered into a text file using a word processor. There is one record for each subject who has participated in an experiment. Each record is made up of the participant's identification number and the number of the experiment in which they participated. In our system, the record keeper enters this file using a word processor on a microcomputer, then uploads it to the VAX system as an ASCII file.

Since the participation list is the link that associates the subjects with the experiments that they have participated in, it is important that this information be as accurate as possible. Customized software was written in order to screen these records prior to being entered into a data table. The program checks each record to see if the student identification number exists within the list of students. A list of nonmatches is provided to the record keeper, who attempts to determine if the nonmatch is the result of an error or if it really represents a person who is not officially a part of the subject pool. Also, the program checks if each entry is unique by comparing it with all previously entered records. This ensures that subjects receive credit for each experiment only once.

If new experiments have been made available since the last update, the ASCII file containing experiments and their credit values is modified using a word processor or editor. The ODL is then used to recreate the experiment data table.

Reports

The design of the subject-pool record-keeping system must ultimately reflect what is required of it. We needed different reports for students, faculty who give credits to student participants, the subject-pool administrator, the experimenters, and a system record keeper. In as much as only the subject-pool record keeper should require a copy of every report, specialized reports present only the

information necessary to the individual who requests it. The decisions about what information to present and the form of presentation were based on interviews with faculty, students, experimenters, and record keepers; modifications to help solve problems over the years the system has been established. The basic information needed in the various reports is shown in Figure 3.

Students. The report for the students consists of multiple alphabetically organized lists. The lists are created by breaking down the students according to class or section. Each list contains names, group identification, the total number of credits earned, and a list of the specific experiments in which each student participated. Since our system rewards students on the basis of time spent participating in experiments and not on the basis of the number of experiments completed, the lists must indicate the total time (or credits) students have completed. These reports are posted in a public place so that students have access to them.

Faculty. Faculty need to know how many credits students have earned in order to reward them. The faculty report is an alphabetical listing of all students in that faculty member's class or group. It lists students' names, students' identification numbers, and the total number of credits or experiments that each student has completed. The identification number is included in the faculty report for a variety of reasons, including differentiation between students with the same name.

Experimenter. Experimenter reports contain the total number of subjects that have participated in each project.

Administration. The administration report contains statistical information relating to the subject pool as a whole. It indicates the participation in each experiment and class.

Record Keeper. The record keeper gets at least one copy of each report generated, so if one is lost, there is a backup copy. The record keeper also receives a master list, one that contains a complete list of all of the participating subjects in alphabetical order, along with their class or section numbers, their identification numbers, the total number of credits or experiments they have completed, and a list of the specific experiments in which they have participated. The record keeper also receives a list of the active experiments and their values for auditing purposes.

Editing

The primary purpose of editing in the subject-pool record-keeping system is error correction. In the simplest terms, editing refers to the ability to add, delete, or modify any specific record of information in any of the files. The ability to edit any portion of the subject-pool record-keeping system is vitally important for several reasons. Editing offers an alternative method of data entry (e.g., to include students who add a class late or add new experiments) and can be used to delete or modify unnecessary or out of date records (e.g., participation entry errors). The editing can, of course, be done on the ASCII files; ODL is then used to transform the information to data tables. However, this editor operates directly on the ORACLE data tables. In some cases (e.g., adding stu-

dents to the student data list), this procedure may be preferable.

Operation in Practice

Although the subject-pool record-keeping system is complex, operating it is quite easy. The operator need not understand the intricacies of ORACLE or SQL programming to be able to use this system. The bulk of the operator's efforts are spent typing in the experiment participation data using his/her choice of word processor. When he/she is finished, this information is saved to an ASCII file.

An effort was made to keep the operation commands simple. A single command is used to initialize and load the system with student information at the beginning of a semester. Updating is also initiated using a single command. The program prompts the operator for the name of the ASCII file to be processed. The sequence of programs is completed without any additional intervention. The report-generation process is equally simple for a user. Again, only a single line command needs to be typed. Because this process takes a while to run, we have fallen into the routine of setting the computer up to execute this sequence of programs early in the morning, when user activity is low. In the morning, all of the reports are automatically generated and printed. However, the short user's manual¹ we created describes both immediate and delayed program execution.

DISCUSSION

We have developed a subject-pool record-keeping system based upon ORACLE on a VAX system. The programs require three ASCII files: a student list, a participation list, and an experiment list. Data tables are created from these ASCII files. A wide variety of reports are generated by combining the information from the various data tables.

This system, including customized software, has been used on line and error-free since January 1991. It was run concurrently with our old record-keeping system for one semester. The content of the output of the two systems was identical, at least with regard to the information both systems could print. Furthermore, we have kept a complete paper-trail backup.

One of our ASCII lists is obtained by downloading from the administrative computer, whereas the others are manually entered. We chose ASCII format to keep great flexibility with regard to the source of these files and to simplify learning to use this system. The record keeper does not have to learn to use a new editor. It is also consistent with advanced applications, such as on-line experiment registration and on-line credit systems.

The reports that we have described are created as ASCII output files that are easy to print. This was the most economical way to provide the experimental participants with feedback. However, since the version of ORACLE that we are using resides on the university computer, it is ac-

cessible by more than one user at a time. Carefully constructed "view" programs could be written through ORACLE, allowing an individual (e.g., student or experimenter) to access information relevant to them (Pepin, 1989).

We designed this system to be able to adapt to future needs. For example, other forms of information can be incorporated with the lists. We have already included e-mail addresses of all students into our student list. E-mail addresses can be used to provide individual electronic reports to students. However, we have not yet implemented this mailing program. We have implemented a feature that automatically adds the date of participation to each record within the participants list. This variable can provide useful statistical information about the subject pool.

The use of ORACLE as the foundation for a subject-pool record-keeping system provides a variety of possibilities other than those that we have discussed. Some features of ORACLE that we have not implemented include the ability to encrypt data for security purposes, data-table indexing to increase information-retrieval speed within fixed data tables, database networking, and exporting databases to other systems (Pepin, 1989).

REFERENCES

- BURNS, J. L. (1974). Some personality attributes of volunteers and of nonvolunteers for psychological experimentation. *Journal of Social Psychology*, **92**, 161-162.
- DAVIS, J. R., & FERNALD, P. S. (1975). Laboratory experience versus subject pool. *American Psychologist*, **30**, 523-524.
- GORNEY, J. (1985). *Invitation to database processing*. Princeton, NJ: Petrocelli Books.
- HARTER, S. P., & PETERS, A. R. (1985). Heuristics for online information retrieval: A typology and preliminary listing. *Online Review*, **9**, 407-422.
- JACKSON, J. M., PROCIDANO, M. E., & COHEN, C. J. (1989). Subject pool sign-up procedures: A threat to external validity. *Social Behavior & Personality*, **17**, 29-43.
- LINDSAY, R. C. L., & HOLDEN, R. R. (1987). The introductory psychology subject pool in Canadian universities. *Canadian Psychology*, **28**, 45-52.
- LIPTON, J. P., & GARZA, R. T. (1978). Further evidence for subject pool contamination. *European Journal of Social Psychology*, **8**, 535-539.
- MILLER, A. (1981). A survey of introductory psychology subject pool practices among leading universities. *Teaching of Psychology*, **8**, 211-213.
- PEPIN, D. (1989). *ORACLE programmer's guide*. Carmel, IN: Que Corp.
- RISHE, N. (1988). *Database design fundamentals: A structured introduction to databases and a structured application design methodology*. Englewood Cliffs, NJ: Prentice Hall.
- SIEBER, J. E., & SAKS, M. J. (1989). A census of subject pool characteristics and policies. *American Psychologist*, **44**, 1053-1061.
- TRICE, A. D., & BAILEY, B. H. (1986). Informed consent: II. Withdrawal-without-prejudice clauses may increase no-shows. *Journal of General Psychology*, **113**, 285-287.

NOTE

1. The record-keeper's manual is a general outline of the procedures used to maintain the system on a periodic basis. It is written for those who have only a basic understanding of computers and software, comparable to word processing knowledge.