Insights into Experiential Pedagogy, Volume 6, 1979

A DATA ENTRY AND RETRIEVAL SYSTEM FOR A COMPUTER SIMULATION (DERS)

David J. Fritzsche, Illinois State University

ABSTRACT

This paper reports the development of a data entry and retrieval system (DERS) which has been created for the purpose of significantly reducing the mechanical tasks associated with administering an educational simulation [1, p. 39]. DERS thus frees up time which the administrator can use to focus upon the creative aspects of the simulation experience. The system also enables the administrator to run the simulation when standard computer input-output devices are not operating. DERS alters the traditional administrative activity in several ways. First, students are responsible for entering their decision into the computer. Students are also responsible for obtaining their decisions from the computer. Second, no computer cards are used in the simulation. All information is put on and retrieved from disc storage. Third, the administrator actually runs the simulation. DERS was created for use with Marketing in Action C2]. However, the structure could be adapted to fit other simulations.

PROGRAM DEVELOPMENT

In its current configuration, DERS consists of three stages. Stage One consists of programs which enable the student team to store company decisions on disc. Stage Two contains the programs which the administrator uses to run the simulation. Stage Three enables the team to retrieve the output of the simulation run upon demand.

Stage One

The heart of the data entry program is a large single dimensional data array. Each element of the array contains one variable of the teams decision. By placing the variables in an array, any variable in the array may be accessed independently of the remaining variables. Thus any variable in the teams decision may be changed without altering the other variables in the decisions.

Each teams decision is divided into company decisions and research requests. When operating in a card environment, the company decisions are punched on one card and the research requests are punched on a second card. This separation is maintained in the data entry programs presentation of decision values to the team members even though both types of data are stored in the same array. When a team accesses the program, it is asked whether it would like a list of the current decisions in its file. The team is then asked whether it would like a list of the current research requests in its file. Samples of the decisions listing and the research requests listing are shown in Tables One and Two.

In addition to the decision and research values stored in the file, Tables One and Two show the variable numbers which are assigned to each variable for identification purposes. Whenever a variable is altered, the variable number must be specified. When a team member refers to a variable number, he is referring to the element of the array in which the value of the variable is stored. After the decision and research values presently on file are listed (if so desired), the team is asked how many variables they wish to change. In order to change a variable, the number of the variable must entered followed by the new value desired for the variable. The new value is then placed in the element of the array referred to by the variable number. Only the variables which the team desire to change need be accessed.

After the decisions for the period have been entered, several provisions are made to check for errors. First, the team is strongly urged to have the program print out their updated decision and research request values. A visual check can then be made to see whether the numbers are those the team intended to enter. Second, any variable value which exceeds the format length of the simulation will appear as asterisks in the decision and research listings. This condition should be discovered and rectified in the visual check of the decision. Finally, several error checks are built into the program. After the team is satisfied that the decision is accurate, the program checks to see whether the sales force allocation sums to one, whether the team is selling a real product, and whether the team has inadvertently fired their sales force. If any of these conditions are found, the program notifies the team and asks them whether they would like to see a listing of their decisions or research requests. The error can then quickly be discovered and corrected.

An abridged flow chart of the data entry program is shown in Figure 1 . It should be noted that the team has the option of listing the decision and research values prior to making any changes, after changes are made, and after an error has been found. If the program finds an error, the error must be corrected before the program will terminate normally. If the error is not corrected, the program will continue to loop.

The data entry program reads the contents of a teams file into computer memory. All of the alterations made to the teams decision and research variables are made upon the values in computer memory. After all of the changes are complete, the new decision is read back into the team's permanent file. After this transfer takes place, the program prints the message "run successfully completed. If this message is not received at the end of a run, the team knows that the program has not terminated normally and thus the decision values in their file have not been updated.

Stage Two

The second stage of DERS consists of a set of programs needed to run the simulation. The administrative data deck containing the parameter values used to run the simulation has been placed on disc and will be referred to as the control file. Prior to running the simulation, a series of activities needs to be accomplished. First the input and Output files must be linked to the simulation program. The simulation program has been slightly altered so that it can read the team decisions from the individual team files. The simulation in its fourth edition can be directed to store the simulation output in individual team files. Second, the current history file is appended to a history storage file. Third, the decisions made by the team are appended to a decision storage file. Fourth, the period number is changed to reflect the current operating period. Finally, the record containing the period number, the record containing any cash adjustments made by the administrator, and the

TABLE 1
DECISION VALUES

The following decision values are currently in your file
(If an "*" appears, the value for that variable is too large)

# 1	2	3
REGION 1.	COMPANY 2.	PERIOD 20.

COLA#	4	5	6	7	8	9	10	11
LEMON	12	13	14	15	16	17	18	19
DIET	20	21	22	23	24	25	26	27

	CARB	SWEET	FLAV	SCHED PROD	PRICE	ADV	SALE FORCE	RET BOT
COLA	2.	2.	2.	535.	3.60	280.	.6	0
LEMON	1.	2.	3.	88.	3.40	130.	.2	0
DIET	2.	2.	4.	151.	3.35	135.	.2	0

# 28	29
TRAINEES 0	SALESFORCE 280

TABLE 2
RESEARCH REQUEST VALUES

The following research requests are currently in your file
(If an "*" appears, the value for that variable is too large)

# 31	32	33
REGION 1.	COMPANY 2.	PERIOD 20.

PAIRED COMPARISON TESTS

	PAIR 1				PAIR 2				PAIR 3			
A#	34	35	36	37	42	43	44	45	50	51	52	53
8#	38	39	40	41	46	47	48	49	54	55	56	57
A	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0

# 58	59	60	61	62	63
		MKT	COMP	COMP	COMP
DEMAND	SEASN	SH	PRICE	ADV	SF
22.	22.	8.	8.	8.	8.

COMPETITOR PRODUCT ANALYSIS

# 64 65	66 67	68 69
FIRM-PROD-A	FIRM-PROD-B	FIRM-PROD-C
0 0	0 0	0 0

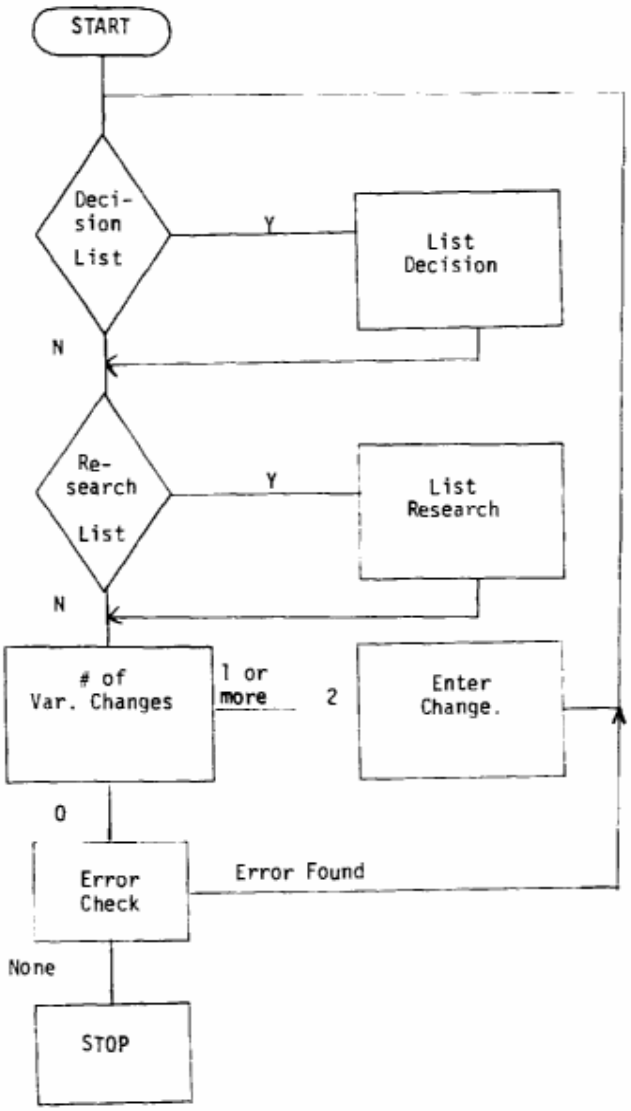
ADV EFFECT	SALES FORCE EFFECT
70 71	72 73 74 75
PROD-BUDG	PROD-SF-#PROD-EFFECT
0 0	0 0 0 0

# 76	77
PREF-PAT	CONT-PREF
0	0

ADV COPY/MEDIA PREF

# 78 79 80	81 82 83
FIRM-PROD-QL-A	FIRM-PROD-QL-B
0 0 0	0 0 0

FIGURE 1
DERS LOGIC



records containing the company decisions are printed at the terminal. These values can be scanned by the administrator and the run aborted if errors are found.

The simulation is then run. After the simulation run is completed, the administrative summary, the last page of the administrators output, is printed on the terminal. The total administrators output is also available on disc. However, reference is generally made to only the last page and thus much paper is saved if only the last page is printed.

When running the simulation on the Mid-Illinois Computer Cooperative's Cyber 72, Stage Two is activated with one command from the administrator: -MIA] (PD 14). The instructions needed for file manipulation and calling the simulation are contained in a procedure file called MIA] with the 1 referring to the region being run. The statement in the parentheses indicates that the current period is Period 14.

A fairly extensive set of disc files is required to run the simulation in a non-card environment. First two files must be allocated for each team, one for the team's decisions and one for the teams output. A file must he created for history storage. A file is needed for the administrative data deck. A file must also be reserved for the administrators output from the simulation. Two files are required for backup data, one for history and one for team decisions. One file is required for the simulation program. Finally, a file is needed for the data entry program, and a file is required to control Stage Two operations. Altogether twenty files are required to operate one region containing six teams. Eighteen files are required for each additional region as the simulation program, and the data entry program are available to use in all regions.

In addition to the above files which are contained in the administrator's account, two procedure files have been created in the student team account. These files contain short programs which simplify accessing the data entry program and the print program. These two files are available for all teams to use.

Stage Three

The third stage consists of a relatively simple procedure file. The file contains a program which enables the team to direct the data in its output file to the line printer. Multiple copies may be made so that each member of the team can have a copy, if desired. Alternatively the team may get copies of their output file by simply listing the file contents on a terminal. The administrator may also obtain a complete copy of his output by directing the system to route the contents of his output file to the line printer. This output is not normally listed on the terminal due to its length.

POTENTIAL PRO8LEMS

There are two potential problems which can occur when using DERS. First, there are no card decks which can serve as a backup in case a file is destroyed. It is imperative that backup files are created for critical data. These include the contents of the history file and the team decision files. In its present configuration, DERS contains files which store the history and the team decisions for all past periods. Thus the present period data can always be restored if a file is destroyed. One would also be wise to make tape copies of all the programs in DERS so that a system crash would not bring you to your knees.

A second problem is one of security. Unless proper precautions are taken, it is likely that unauthorized proprietary information will leak into competitors hands. The way this problem is handled will depend to some extent upon the computer system available. At Illinois State, all of the teams files as well as the simulation and the DERS programs are stored under one account. The teams access the computer using another account. Each team is given read and write permission for its input file and read permission for its output file and the data entry program. The read and write permission for the input and output files is contingent upon the proper password which is supplied to the individual teams. The teams are informed that the password is the key to their files and thus should be carefully protected. If a team believes their password has been discovered by a competitor, it may request a new password from the administrator.

By keeping all of the files in an account not accessible to students except by appropriate password, there is no opportunity for teams to damage the data entry program or their files either by accident or on purpose. Their contact with their files is through procedure files, never direct. The procedure files are in the student account; however, they are very simple and easy to replace if damaged. To date, this has not been necessary.

USE EXPERIENCE

So far the use experience has been quite positive. Students tend to feel that the terminal is more interesting to use than the key punch. They like to be able to pick up their output at their convenience. The number of errors associated with the simulation decisions has been significantly reduced. This is due to the fact that the team has a chance to examine their decision after it has been entered into memory and that the data entry program checks for specific errors and when found will not let the team terminate the program until the errors are corrected.

From the administrators perspective, DERS has reduced the amount of time required to process a decision. Instead of collecting and checking cards for errors, submitting cards for a run, and distributing the output, the task involves typing one instruction on the terminal and then a scanning of the decisions previously checked by the computer.

There are several distinct operational advantages to DERS. First, the administrator has complete control over the simulation run. He does not have to depend upon computer center personnel. Second, if the card reader or printer is out of order, the simulation is not affected. The output can be obtained via terminal. Third the simulation play does not have to revolve around class meetings. Times can be set for decisions to be due. The decision can then be run with the output being picked up at the teams convenience. Thus more than one decision can be made through the week even though a class may meet only one night a week. This configuration is also advantageous if the administrator is ill. The simulation can be run from the bedside if a portable terminal and phone are available.

Possibly one of the biggest advantages of DERS is that it forces the student to become more comfortable with the computer via familiarity. The computer terminal must be used to participate in the simulation. It becomes a tool which is used to accomplish a task. Thus, if the student finds a terminal on the desk when assuming the first job out of college, no trauma is likely to occur.

REFERENCES

[1] Fritzsche, David J., “Lets Simplify the Administrative Requirements of Computerized Educational Simulations,” Exploring Experiential Learning: Simulations and Experiential Exercises, The Proceedings of the Fifth National A8SEL Conference, Denver, Colorado, 1978, pp. 39-43.

[2] Ness, Thomas E. and Ralph L. Day, Marketing In Action: A Decision Game, 4th ed., (Homewood: Richard D. Irwin, Inc., 1978).