

# Developments in Business Simulation & Experiential Exercises, Volume 9, 1982

## FILE ACCESS IS THE ESSENTIAL PREREQUISITE TO TIME-FLEXIBLE AND INTERACTIVE COMPUTER SIMULATION

Newell E. Chiesl, Indiana State University

### INTRODUCTION

The of this paper is to illustrate the methodology of time-flexible interactive business games. The importance of the work is to illustrate how to develop time-flexible interactive computer simulations so that more people are able to use their own computer games of this type in the classroom.

A time-flexible interactive business game offers the student a more realistic environment than the fixed-time format business game. This characteristic, referred to as verisimilitude, means that the participants experience a business environment that has the appearance of being true and real. This is the salient characteristic that makes this type of computer simulation mandatory for use in the classroom while making the fixed-format games obsolete in comparison.

The methodology utilized by this paper to accomplish the above stated purpose will be to: 1) define the nature of time-flexible interactive simulation, 2) present a file access and modification example, and 3) in a very basic way, elucidate the technical aspects of the programming.

#### The Nature of a Time-flexible Interactive Simulation

The time-flexible and double interactive gaming technique is a computer simulation where participants 1) select their own time format, and 2) interact with the computer system and other participants. The first way the double interactive technique is interactive is by allowing the participants to interact with the computer system by viewing the results, output of their decisions on a cathode ray tube (CRT) or Digital DECWRITER TYPE terminal. To users (participants) of a time sharing system, it appears that their requests are instantaneously answered. In reality the main computer has probably answered a dozen or more requests to other interactive terminal users. [1]

The second way the users interact is by interacting with other persons participating in the simulation game. That is to say, the amounts of the variables selected by participants #1-9 have an influence on the results of participant #10.

The double interactive nature of gaming techniques is analogous to playing The Star Trek Game not just against the heuristically thinking computer, but against persons using the terminals and interacting with your playing the roles of the KLINGONS and ROMULINS.

Time flexibility relates to the length of time being simulated when a decision is inputted. In fixed time period format games the participant's decisions remain in effect for the same amount of simulation time for each computer run. For example, in many games the decisions are made quarterly during the year. The compressed simulation time, then, is always three months for this type of game. In a time flexible interactive game, participants are allowed to select the amount of time the decisions are to be simulated.

For example, participants could allow their decisions to be simulated for 2.5 weeks, 27.563 weeks, or even .2 weeks. This compressed simulation time could correlate directly to the wall clock time (not CPU time) of 2.5 minutes, 27.563 minutes and .2 minutes. This flexibility allows students to learn by experimenting with their input variables while monitoring the output variables.

Participants are able to select the length of time their decisions are simulated. The method is implemented by allowing the appropriate wall clock time (that corresponds to the compressed simulation time) to expire and then request profit and loss statistics. The method is real-time dependent and relies on wall clock time being elapsed. Because of the interactive nature of the gaming technique, it is imperative that all participants initiate the simulation at as close to the same time as possible. A couple of minutes of variance is acceptable. However, when the variance is too large, it is possible that one participant will be experiencing a negative exogenous phenomenon in the scenario which will adversely affect the choice of inputs, while another participant at the same moment in time is experiencing a pleasant decision making environment. Since the collective action of the group's decisions have a negative, positive, or neutral influence on each participant, it is necessary to rely on the elapsed wall time to govern the compressed simulation time for every participant.

An example will illustrate this reliance on wall clock time. In a business oriented interactive game, it is necessary to have all participants at the same time experiencing the affects of such uncontrollable factors as inflation, business cycles, introduction of a substitute product, government intervention in pricing, or a change in consumer preference. The importance of the wall clock reliance is readily seen if participants #1-9 adjust their decision price variable to the \$130- \$140 range in reaction to scenario event N in the simulation while player #10 at the same time markets the product at \$180 while experiencing event T in the scenario. The price variance would, of course, significantly lower the demand for participant #10's product.

#### File Access and Modification

The hardware required for the time flexible and Double Interactive Gaming technique consist of a main computer, interactive terminals with I/O display and an On-Line disc storage system which has indirect access capabilities.

The previous part of this paper discussed the operational characteristics of a defined interactive gaming technique. This section illustrates the technique involved in accessing and modifying the data files in the simulation.

The computer system used for this business simulation is a CDC Cvber 171-2 using NOS 1.4 and NAM/IAF. Therefore, the terminology utilized in this paper is peculiar and indigenous to this particular system.

## Developments in Business Simulation & Experiential Exercises, Volume 9, 1982

Those familiar with other systems will, of course, be versed with other terminology. It is also recognized that it is possible to do certain tasks, such as file access, on one computer system that are impossible to accomplish on another system. The important part of this paper is not technical, it is pedagogical. With this in mind, it is a fairly easy task to substitute the terminology from the example system to your system.

Figure I is a very simplistic diagram of the events through time. For ease of illustration, two participants will be diagrammed; after all, the author is only human. Diagramming more than two participants will not significantly increase the illustrative effect, but will exponentially increase the difficulty of the diagram's construction. Interactive Gaming techniques are able to accommodate any number of participants and data files. The only limitations are the number of terminals, amount of disc storage available, and, of course, the tenacity of the programmer.

Referring to Figure I, participant #1 starts a marketing game by simulating the initial values of the variable given at time period A. At time period H a modification is made in the decision variables. A modification or change is made in one of the decision variables by calling up the appropriate data file. The participant of the interactive gaming technique needs no programming knowledge; the program does all the details. The data file is accessed indirectly. Indirect access means that the program receives a copy of the data file, the actual data file remains in disc storage. Indirect access files can be read by many users at the same time since the original always remains in disc storage and is not removed. Direct access files can only be read by one user at a time. Any changes made in the data file are made directly in disc storage. This is unlike indirect access, where it is possible to change, modify, or purge a scratch copy of the data file without modifying the original file. When the modification of the scratch copy is complete, a separate program request is made (called) to the system to replace the original indirect data file with the newly modified file. It is important to understand the indirect nature of the call routine. Without this feature a truly interactive and time flexible gaming technique would not be possible.

Referring again to Figure I, participant #1 continues until a read request is called at time period L. The read request is a preliminary call made before an output generation. It serves the function of gathering the collective data by reading all of the participants' data files and then summarizing the results in the form of averages. As was noted earlier in this paper, the collective decisions of the group have a positive, negative, or neutral affect on each participant. These averages, although not seen by participants #1, are used in the mathematical subroutines which calculates the results (output) of the simulated time. The output, in time period M, is then routed to the I/O terminal, from which participant #1 views the output of simulated time A-M.

Participant #2, in Figure I, starts the interactive game in a different, but more analytic manner. A request is made for information in time period B. The program responds by reading the indirectly accessed files of the other participants.

Summary statistics are then made available to participant #2. After analysis, it is determined in time period C to modify participant #2's decision inputs. Time continues until the simulated outputs of time B-J are routed to the terminal.

It should be noted that the procedure, utilized to read the group files before an output generation, is exactly the same as the procedures employed when a participant requests information (in the form of group averages) from all of the other participants. Although the same access procedure is used, the participant does not see the summary statistics in the read-output mode.

Participant #2 continues the simulation by requesting information, such as marketing research, in time period K. Participant #2, after determining that their decisions do not need to be modified, continues with those decisions until the end of simulation. Performance for the two participants is then evaluated by the game administrator. Usually for a marketing game, profit and market share are the criteria.

Through the use of the above example, it has been seen that participants are able to select any amount of time they wish their decision to be in affect. In Figure I, therefore, participant #1 receives a profit and loss statement for time periods A-H, H-N, and M-Z. Participant #2 also receives profit and loss statements, but this player has their decisions in affect for different time periods, A-C, C-J, and J-Z. This feature of allowing participants to change decision inputs at any time truly results in the participants' playing a very realistic computer game.

### Programming Example

To simplify the demonstration of the FORTRAN program, only three basic marketing decision modules will be presented. In practice, computer games are able to easily accommodate twenty or more complex modules.

Figure II presents the flowchart of the program. Participants are able to select one of three modules. When Information is selected, the game player obtains current market information on competitors' average expenditure on newspaper advertising. Current information is exactly that, up to the latest millisecond. Without indirect file access, a participant would only obtain the same information stored in the file as it appeared at the compiling stage of the program. The information would never change, if this were the case, and no matter how many times a participant made information requests the information result would be identical. Fortunately, this is not the case and current Information is available.

Now how does this happen? The initial step is for the business game designer to determine the desired learning objectives, formulate the associate mathematical relationships and write the computer program. Inside the program will be a call routine to store the decision variables in a file, say called NA1. The next step is to run the program and determine if all the quantitative routines in the program perform as intended. After this step, the same program is duplicated until ten identical programs are obtained, assuming the game administrator wants ten competitors in the market place. (Each competitor could actually be a group of 1-5 students.)

The final step is to go into each program and edit the file names to match the designated player's number. For instance, participant #5 runs program 5 with the file name NA5. (NAS for newspaper advertising #5.) When participant #5 modifies their amount of advertising, the amount in file NAS will immediately change. In computer talk (FORTRAN), program #5 looks something like the following:

## Developments in Business Simulation & Experiential Exercises, Volume 9, 1982

```

4  IF (T.EQ.Z) GO TO 1000
    PRINT *, "WHAT DECISION MODULE DO YOU SELECT"
    PRINT *, "INFO=1; NEWSADV2; P&L=3"
    INPUT *, A
    IF (A.EQ.1) GO TO 1
    IF (A.EQ.2) GO TO 2
    IF (A.EQ.3) GO TO 3
    GO TO 4
1  CALL GET ("NA1")
    READ 11, X(1)
    CALL GET ("NA2")
    READ 12, X(2)
    CALL GET ("NA3")
    READ 13, X(3)
    CALL GET ("NA4")
    READ 14, X(4)
    CALL GET ("NA6")
    READ 16, X(6)
    CALL GET ("NA7")
    READ 17, X(7)
    CALL GET ("NA8")
    READ 18, X(8)
    CALL GET ("NA9")
    READ 19, X(9)
    CALL GET ("NA10")
    READ 20, X(10)
    DO 40 I=1,9
    AVNA=X(I)
40  CONTINUE
    AVNA=AVNA/ 9
    PRINT *, "AVERAGE COMPETITORS' AMOUNT SPENT ON
    NEWSADV PER WEEK
    PRINT *, AVNA
    GO TO 4
2  CALL GET ("NA5")
    READ 15, NA#5
    PRINT *, "THE CURRENT AMOUNT SPENT PER WEEK ON
    NEWSADV
    PRINT *, NA#5
    PRINT *, "DO YOU WISH TO CHANGE THIS AMOUNT"
    PRINT *, "YES=1; NO=2"
    READ *, A
    IF (A.EQ.1) GO TO 31
    GO TO 4
31 PRINT *, "WHAT IS THE NEW AMOUNT?"
    READ *, NA
    CALL GET ("NA5")
    REWIND 15
    WRITE 15, NA#5
    ENDFILE 15
    REWIND 15
    GO TO 4
3  PRINT *, 'P&L FOR PARTICIPANT #5"
    CALL SUBROUTINE AVERAGES (AVNA)
    AE1=HD - (HD-(HD*(12.71828)**
    (-NA#5 (KU))
    AE2=K2*(NA#5/AVNA)
    AE=AE 1+AE2
    PRINT *, "NEWSADV AFFECT
    PRINT *, AE
    GO TO 4
1000 STOP
    END

```

Notice that the first part of the program asks the participant which module they wish to select. Then if

information is requested, the program calls and reads files 1-4 and 6-10. This procedure reads the current contents in those files and then takes an average of the NEWSADV, variable AVNA. Therefore what is actually occurring is that participants #1-10 are in a constant communication with each other by leaving messages in a "drop-box" type of arrangement. In computer talk this feature is named indirect access for data files and call routines inside the FORTRAN program.

Participant #5 might request to change their amount of dollars spent on newspaper advertising per week, variable name NA#5. Program 5 calls and gets file name NA5 which stores the amount spent, asks for the new amount and modifies file NA5 by writing in the new amount. Immediately, the other participants #1-4 and 6-10 are able to obtain this new amount.

Participant #5 will now request a P&L statement, which for our purposes is advertising effectiveness, variable name AE. The program starts by going to 3 and then calling subroutine averages that calculates the average amount spent by competitors on newspaper advertising, AVNA. AE1 is the name of the variable that calculates part of the advertising's effectiveness. This formula equates to Figure III where the AE1 curve becomes asymptotic to the highest amount of demand obtainable (variable name HD) as more and more money is spent on newspaper advertising. AE2 is the second part of determining the total AE and is simply a ratio times a constant, variable K2.

The time-flexibility feature is directly related to clock time. Everytime a P&L statement is requested, the program determines internally the time. It is a simple procedure to then subtract one time from another. The result is the elapsed time from one P&L request to another. If for example the result is 6.54 minutes, the compressed simulation time, for simplicity sake, could be 6.54 weeks, months, or years. Participants are informed of this feature before the game starts.

Question: If newspaper advertising is an amount per week and a participant simulates 6.54 weeks and then changes the rate for the next P&L statement, it must be impossible to calculate demand in units? Answer: No, the entire game thinks in rates per week and adjustments are made in the programming. For example, if participant #5 is spending \$100 per week for 6.54 weeks, then \$654 is spent on newspaper advertising for the amount of simulation time for the P&L statement.

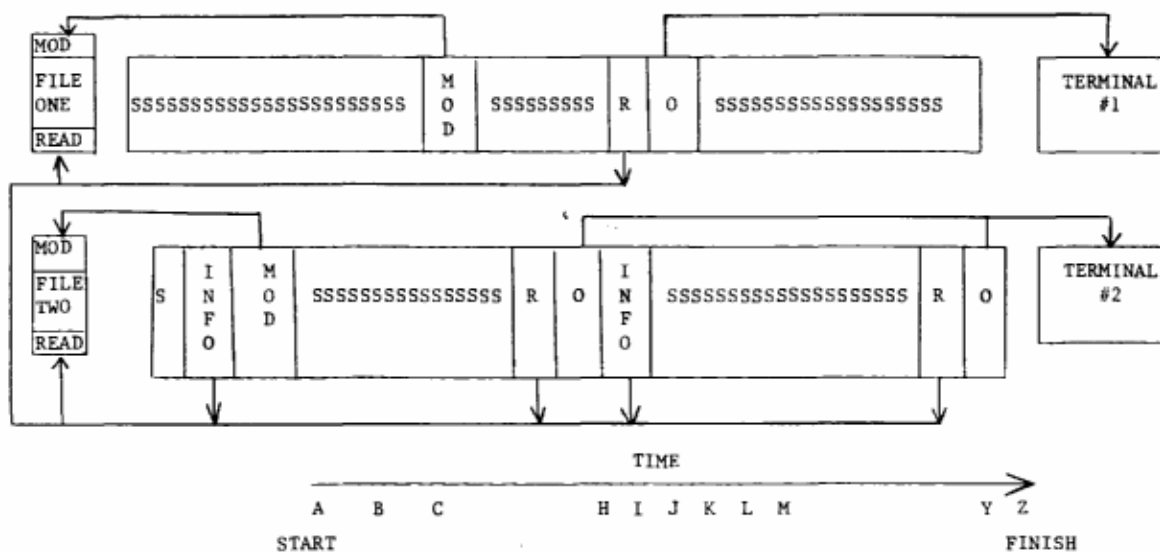
### Summary

This paper has been designed to familiarize more people with the nature and methodology of time-flexible interactive computer simulations. A related purpose has been to illustrate the dynamic and very realistic nature of time-flexible games as compared to the fixed format games. It is hoped that more users of computer games will adopt this new technique in business simulations.

### REFERENCES

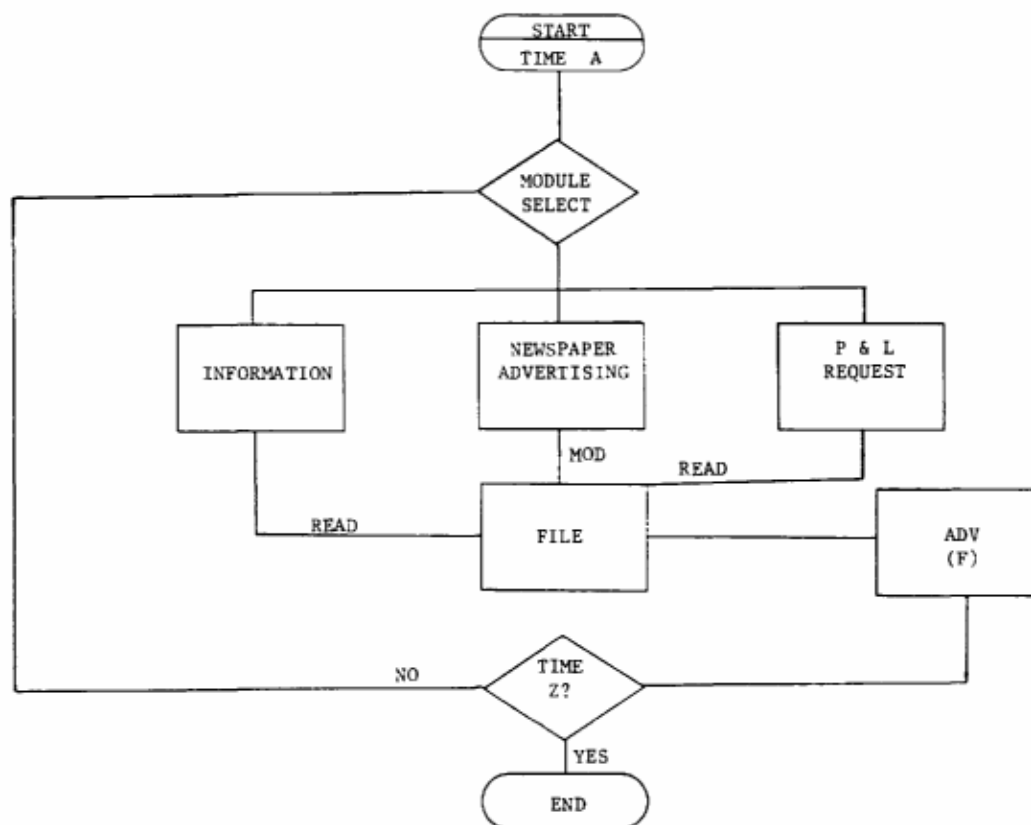
- [1] Chiesl, Newell, "The Dynamic Aspects of Interactive Gaming Puts the Realism into Gaming." Insights Into Experiential Pedagogy. Proceedings of the Sixth Annual Conference of ABSEL (1979), pp. 271-276.

FIGURE I  
FILE ACCESS AND MODIFICATION DIAGRAM  
(For Two Participants, One File Per Participant)



CODE	
S	Simulation Time
Info	Information Request
MOD	Modification
R	Read
O	Output

FIGURE II



## Developments in Business Simulation & Experiential Exercises, Volume 9, 1982

FIGURE III  
NEWSPAPER ADVERTISING EFFECTIVENESS

