2

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

THE DESIGN AND DEVELOPMENT OF A GAMING INTERFACE FOR THE SYSTEM DYNAMICS MODEL OF SOFTWARE PROJECT MANAGEMENT

by

Edward Tulenko

March 1989

Thesis Advisor: Tarek K. Abdel-Hamid

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT
Approved for public release; distribution is unlimited

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
Naval Postgraduate School

6b. OFFICE SYMBOL *(If applicable)*
Code 54

7a. NAME OF MONITORING ORGANIZATION
Naval Postgraduate School

6c. ADDRESS *(City, State, and ZIP Code)*
Monterey, California 93943-5000

7b. ADDRESS *(City, State, and ZIP Code)*
Monterey, California 93943-5000

8a. NAME OF FUNDING/SPONSORING ORGANIZATION

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State, and ZIP Code)*

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO

11. TITLE *(Include Security Classification)*
THE DESIGN AND DEVELOPMENT OF A GAMING INTERFACE FOR THE SYSTEM DYNAMICS MODEL OF SOFTWARE PROJECT MANAGEMENT

12. PERSONAL AUTHOR(S)
Tulenko, Edward

13a. TYPE OF REPORT
Master's Thesis

13b. TIME COVERED
FROM ________ TO ______

14. DATE OF REPORT *(Year, Month, Day)*
1989, March

15. PAGE COUNT
67

16. SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Software Project Management; System Dynamics; Gaming Interface ; ([illegible]

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

Software project managers have been plagued in the software development process with an infamous reputation for cost overruns, late deliveries, poor reliability and users' dissatisfaction. The Dynamica Model of Software Project Management has been designed to support the management of the software development process.

The objective of this thesis is to enhance the user interface to the Dynamica Model of Software Project Management by incorporating Gaming. More specifically, software project managers will be able to stop a simulation of a software development project at different intervals, assess project status, and react by altering project variables in real time. This mirrors the dynamic decision making process that software project managers experience in a real world environment. Keywords; [illegible] Military theses;

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED □ SAME AS RPT. □ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Prof. Tarek K. Abdel-Hamid

22b. TELEPHONE *(Include Area Code)*
(408) 646-2686

22c. OFFICE SYMBOL
Code 54Ha

83 APR edition may be used until exhausted
All other editions are obsolete

☆ U.S. Government Printing Office: 1986—606-243

The Design and Development of a Gaming Interface for the System Dynamics Model of Software Project Management

by

Edward Tulenko
Lieutenant, United States Navy
B.S., Pennsylvania State University, 1983

Submitted in partial fulfillment of the requirements for the degree of

MASTERS OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
March 1989

Author: Edward Tulenko
Edward Tulenko

Approved by: T. Hamid
Tarek K. Abdel-Hamid, Thesis Advisor

Magdi Kamel, Second Reader

David R. Whipple, Chairman
Department of Administrative Sciences

K.T. Marshall
Kneale T. Marshall,
Dean of Information and Policy Sciences

ABSTRACT

Software project managers have been plagued in the software development process with an infamous reputation for cost overruns, late deliveries, poor reliability and users' dissatisfaction. The Dynamica Model of Software Project Management has been designed to support the management of the software development process.

The objective of this thesis is to enhance the user interface to the Dynamica Model of Software Project Management by incorporating Gaming. More specifically, software project managers will be able to stop a simulation of a software deve` ɔpment project at different intervals, assess project status, and react by altering project variables in real time. This mirrors the dynamic decision making process that software project managers experience in a real world environment.

TABLE OF CONTENTS

I. INTRODUCTION

A. BACKGROUND

In recent years, rapid technological advancements in computer hardware, and the ensuing cost reduction of equipment, has increased the demand for hardware and consequently the demand on software. A tenfold increase in software demand is expected over the next 10 years [Ref. 1:pp. 55-62]. Such a tremendous growth in the software industry creates numerous problems for software project managers: cost overruns, late deliveries, poor reliability and user dissatisfaction [Ref. 2:pp. 36-41] and [Ref. 3:pp. 132-142]. Only recently has the software project manager seen the development of an assortment of "tools" to aid him in estimating, tracking and forecasting costs, scheduling completion dates, and in numerous other tasks which are integral parts of the software development process.

The Dynamica Model of Software Project Management is one of the exciting new "tools" recently developed [Ref. 4:pp 8-12]. It is a comprehensive model of the software development process. The model, written in Professional Dynamo, integrates both the management type functions (e.g., planning, controlling and staffing) with the software production type activities (e.g., design, coding, reviewing and testing).

The Dynamica Model of Software Project Management can perform several important roles. Its main goal is to aid the software project manager in understanding the software development process. The model allows the user to track, store, graph and plot large amounts of project data, quickly and efficiently. The manager can then conduct "what if" experiments with the model to develop a more comprehensive understanding of the interrelationships of software development variables. As a result, the user improves his fundamental understanding of the software development process through utilization of a computer simulation model [Ref. 5:p. 2].

Secondly, the Dynamica model can be used to aid the software project manager in the actual management process. The model can be utilized to estimate project cost, schedule completion time, and numerous other variables in a software development project. The manager can alter these variables, run the simulation and view results for analysis in a matter of minutes.

B. PURPOSE OF RESEARCH

The objective of this thesis is to enhance the user interface to the Dynamica Model of Software Project Management by incorporating Gaming. More specifically, software project managers will be able to stop a simulation of a software development project at different intervals, assess project status, and react by altering project variables in real time. This mirrors the dynamic decision making process that software project managers experience in a real world environment.

C. SCOPE OF RESEARCH

The scope of this research will include the design and development of a Gaming interface for the Dynamica Model of Software Project Management. The Gaming interface will utilize Dynex (DYNAMO for Executives) and Professional Dynamo's Gaming Facility.

D. THESIS ORGANIZATION

Chapter II briefly discusses some problem areas in the current methods of software project management and the role of the Dynamica Model of Software Project Management to solve those problems. Chapter III provides two Gaming examples for analysis. Chapter IV discusses the system architecture of the Gaming Facility created in this thesis.

II. RESEARCH BACKGROUND AND OBJECTIVES

A. CURRENT PROBLEMS IN SOFTWARE PROJECT MANAGEMENT

There has been tremendous growth in the demand for software systems over the past 20 years. The software development process, unfortunately, has earned an "infamous" reputation for cost overruns, late deliveries, poor reliability and users' dissatisfaction [Refs. 2:pp. 36-41; 3:pp. 132-142].

While significant progress has been made over the past 20 years in improving the technology of software development, little research effort has been devoted to the managerial issues.

> Software Engineering Project Management (SEPM) has not enjoyed the same progress (as the technology of software development). While it might be argued that SEPM has been defined, it is far from a recognized discipline....The major issues and problems of SEPM have not been agreed on by the computing community as a whole, and consequently, priorities for addressing them have not been widely established. Furthermore, research in this area has been scant. [Ref. 6:p. 333]

B. THE DYNAMICA MODEL

The goal of the Dynamica model is to provide an understanding of the dynamic behav1or of software projects and support the management of the software development process [Ref. 4:pp. 8-10].

The Dynamica model is a comprehensive system dynamics model of the software development process. The model integrates the multiple functions of the software development process, including both the management type functions (e.g., planning, control, staffing) as well as the software production type activities (e.g., design, coding, reviewing, testing) [Ref. 7:pp. 6-11]. Such an integrative approach is useful since it would prompt and facilitate a search for the multiple and potentially diffuse set of factors that are interacting to cause some software project problem(s) [Ref. 7:p. 14].

Another distinctive aspect of the Dynamica model is its use of computer simulation techniques to handle the high complexity of the integrative feedback model.

> The behavior of systems of interconnected feedback loops often confounds common intuition and analysis, even though the dynamic implications of isolated loops may be reasonably obvious. The feedback structures of real problems are often so complex that the behavior they generate over time can usually be traced only by simulation. [Ref. 8:pp. 6-7]

The Dynamica model consists of the four subsystems shown in Figure 2-1 [Ref. 4:p. 12]. The Human Resource Management Subsystem captures the hiring, training, assimilation, and transfer of the project's human resources. The Software Production Subsystem captures the design, coding, quality assurance, rework, and testing activities [Ref. 7:pp. 11-25]. The Planning Subsystem models the scheduling activities that take place throughout the project's life cycle. The Control Subsystem captures the measurement of progress on the project [Ref. 5:pp. 6-8].

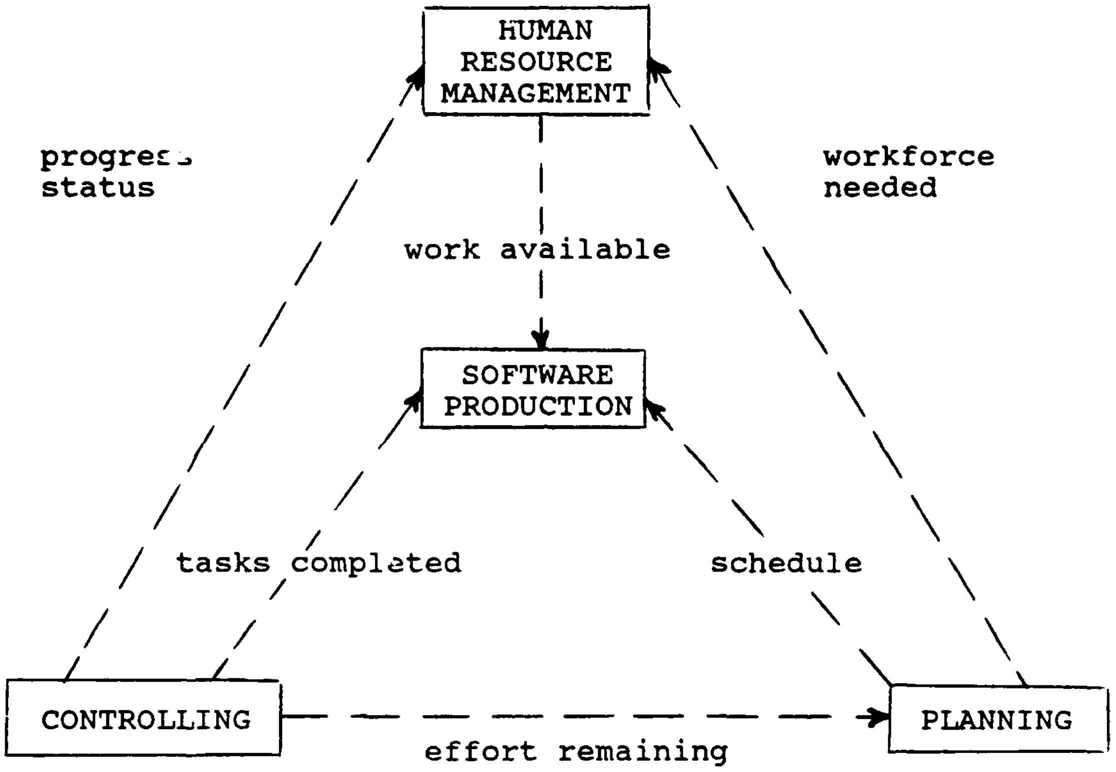


Figure 2-1 Four Subsystems of The Dynamica Model

C. RESEARCH OBJECTIVES

A major advantage of the Dynamica simulation model is its capability to provide the user with a dynamic and detailed picture of how model variables change throughout the project life cycle. The major goal of this thesis is to improve this picture by incorporating Gaming. Gaming will add the capability of stopping a simulation at different intervals, assessing project status and reacting by altering project variables in real time. Secondly, this thesis will illustrate the effects of dynamic managerial decision making, using Gaming, through the comparison of two software project development examples.

III. GAMING EXAMPLE

Models serve to develop insight into a complex situation and to train others to share those insights. Once the model is thoroughly debugged and its lessons understood, it can become the basis for a game or training tool. The user (player) is explained the situation by a series of prompts and asked to make several decisions to reach a stated goal. The model is then simulated for a certain period of time and the results displayed. The user is again prompted for his decisions given the updated situation. Once the revised decisions are complete the simulation is resumed and the new results displayed. This cycle of decision making, simulation resumption for a certain period of time, and results display continues until the goal is reached or the results of the decision are clear [Ref. 9:p. 2].

Gaming uses three Professional Dynamo modules, DYNEX, Simulate, and Report. DYNEX displays a series of prompts explaining the situation and then prompts the player to type in values reflecting his management decisions.

Simulate reads the compiled model, the user's decisions, and the conditions that existed at the end of the last simulation period that the game builder specifies.

Report displays the results from the beginning through the last simulation period.

This chapter cites two examples that illustrate the usefulness of Gaming. These examples guide the player through Gaming as would a user's manual. Example One is simulated without the player changing any variables throughout the simulation. In Example Two, the player changes one variable at various time intervals. The results of the two examples are then compared and analyzed to illustrate the effects of the player's decision to alter a project variable. The project variable that will be altered in Example Two is HIRING DELAY (HIREDY). HIREDY is the average delay time, in work days, incurred in adding new staff members to the project. For additional explanations of project variables see [Ref. 5].

A. EXAMPLE ONE

To initiate the Gaming Facility type <CMPL PROJECT> and press <ENTER> at the DOS prompt of the directory containing the files of the systems disk. This compiles the sample model PROJECT.DYN. Next, type <GAME PROJECT> and press <ENTER>. This executes the GAME.BAT file and displays Figure 3-1.

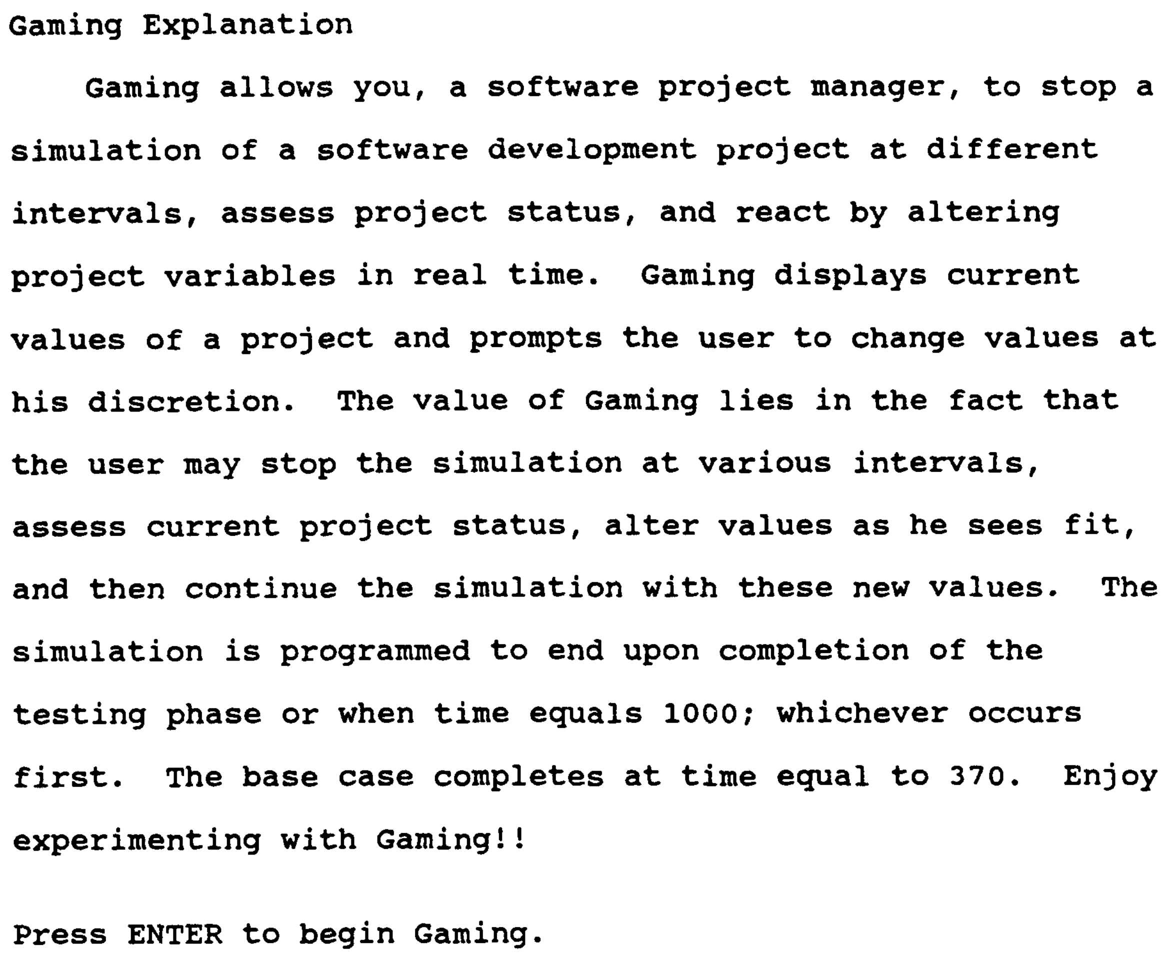
Welcome to Gaming !!

The Gaming Facility introduced here will allow soft-ware managers to input decisions, start a simulation, assess the consequences of those decisions, and adjust model variables dynamically at various intervals throughout the simulation run. This provides a realistic training environment by allowing interaction with the simulation on a continuous basis throughout the software development life cycle.

Press ENTER to see page two of explanation on how to play the game.

Figure 3-1 Gaming Explanation--Page One

Press <ENTER> to see page two of the Gaming explanation as shown in Figure 3-2.



Gaming Explanation

Gaming allows you, a software project manager, to stop a simulation of a software development project at different intervals, assess project status, and react by altering project variables in real time. Gaming displays current values of a project and prompts the user to change values at his discretion. The value of Gaming lies in the fact that the user may stop the simulation at various intervals, assess current project status, alter values as he sees fit, and then continue the simulation with these new values. The simulation is programmed to end upon completion of the testing phase or when time equals 1000; whichever occurs first. The base case completes at time equal to 370. Enjoy experimenting with Gaming!!

Press ENTER to begin Gaming.

Figure 3-2 Gaming Explanation--Page Two

Press <ENTER> once again and the Model Menu for Gaming appears as in Figure 3-3.

```
                    MODEL MENU FOR
   MANIPULATION OF MODEL VARIABLES USING GAMING

          1.  NO CHANGES--SIMULATE

          2.  INTERVAL TO SIMULATE

          3.  ESTIMATED ACTUAL PROJECT SIZE

          4.  ORGANIZATIONAL ENVIRONMENT VARIABLES

          5.  POLICY VARIABLES

Enter the number(s) of your selected choices.
(Separate each choice by a space or a comma.)
```

Figure 3-3 Model Menu

The Model Menu provides the user five topic areas to choose from. Each option will be explained as we walk through Example One of Gaming. Choosing an option simply requires typing the number of the selected choice(s) and pressing <ENTER>. Separate each choice by a space or a comma.

Choosing option one, NO CHANGES--SIMULATE, simulates the base model using all the preset values for the variables. DO NOT choose option one at this time. It will be discussed later in the example.

Choose options two, three, four and five by typing <2,3,4,5> and pressing <ENTER>. This will allow the user to view these four options to better understand Example One. The first screen that is displayed is shown is Figure 3-4.

This is also the screen that appears if the user selects option two from the Model Menu, INTERVAL TO SIMULATE.

SETTING INTERVAL TO SIMULATE

1) Press <ENTER> to accept the preset value

or

2) Enter the new value and press <ENTER>

The preset value of INTERVAL TO SIMULATE = 50.

Figure 3-4 Interval to Simulate

The simulation interval throughout Example One will remain 50. Therefore, press <ENTER>, as prompted, to accept the preset value of 50 for INTERVAL TO SIMULATE.

Next, Figure 3-5 is displayed. This is also the screen that appears if choosing option three from the Model Menu, ESTIMATED ACTUAL PROJECT SIZE.

SETTING ACTUAL PROJECT SIZE VARIABLE

1) Press <ENTER> to accept the preset value

or

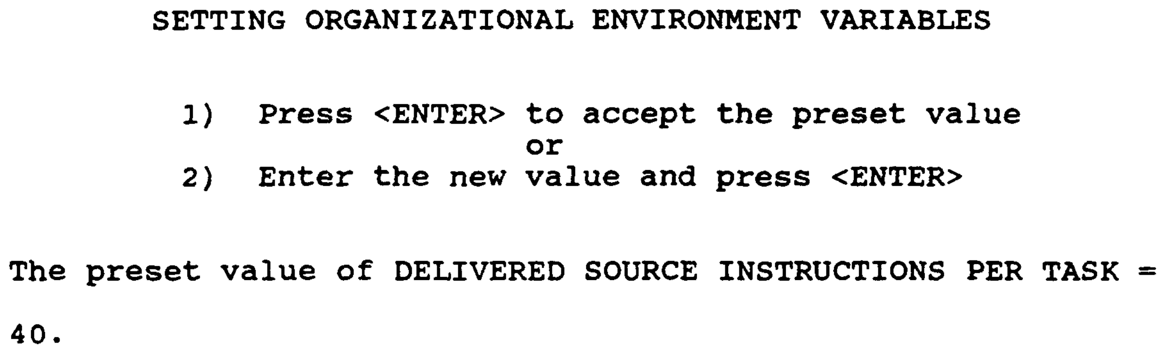
2) Enter the new value and press <ENTER>

The preset value of REAL JOB SIZE IN DELIVERED SOURCE INSTRUCTIONS = 24.4e3

Figure 3-5 Real Job Size in Delivered Source Instructions

The Real Job Size in Example One will be 24,000 Delivered Source Instructions. Press <ENTER> to accept the preset value of 24.4e3 (24,400) for REAL JOB SIZE IN DELIVERED SOURCE INSTRUCTIONS.

The next five screens pertain to option number four of the Model Menu, ORGANIZATIONAL ENVIRONMENT VARIABLES. The first screen, Figure 3-6, shows the value of DELIVERED SOURCE INSTRUCTIONS PER TASK = 40. Press <ENTER> to accept this preset value.



SETTING ORGANIZATIONAL ENVIRONMENT VARIABLES

1) Press <ENTER> to accept the preset value
or
2) Enter the new value and press <ENTER>

The preset value of DELIVERED SOURCE INSTRUCTIONS PER TASK = 40.

Figure 3-6 Delivered Source Instructions Per Task

Next, Figure 3-7 displays the preset value of HIRING DELAY = 30. Here, type <100> and press <ENTER> to change the variable HIRING DELAY from 30 to 100. HIRING DELAY will continue to remain 100 throughout the simulation.

```
          SETTING ORGANIZATIONAL ENVIRONMENT VARIABLES

                1)  Press <ENTER> to accept the preset value
                                  or
                2)  Enter the new value and press <ENTER>

The preset value of HIRING DELAY = 30.
```

Figure 3-7 Hiring Delay

The next three screens display the preset values of ASSIMILATION DELAY=21.0 (Figure 3-8), AVERAGE EMPLOYMENT=1000 (Figure 3-9) and ERROR RATE PER 1000 DELIVERED SOURCE INSTRUCTIONS in a matrix format = 24, 22.9, 20.75, 15.25, 13.1, and 12 (Figure 3-10). Press <ENTER> as prompted at each screen to accept these preset values.

```
         SETTING ORGANIZATIONAL ENVIRONMENTAL VARIABLES

                1)  Press <ENTER> to accept the preset value
                                  or
                2)  Enter the new value and press <ENTER>

The preset value of ASSIMILATION DELAY = 21.
```

Figure 3-8 Assimilation Delay

```
         SETTING ORGANIZATIONAL ENVIRONMENT VARIABLES

             1)   Press <ENTER> to accept the preset value
                                 or
             2)   Enter the new value and press <ENTER>

The preset value of AVERAGE EMPLOYMENT = 1000.
```

Figure 3-9 Average Employment

```
         SETTING ORGANIZATIONAL ENVIRONMENT VARIABLES

             1)   Press <ENTER> to accept the preset matrix
                  values
                                  or
             2)   Enter the new matrix values and press <ENTER>
                  (Values must be separated by a space or comma;
                  To change any value in the matrix you re-enter
                  all values)

The preset values of ERROR RATE PER 1000 DELIVERED SOURCE

INSTRUCTIONS are:

             1      2       3       4      5      6

            24    22.9    20.75   15.25   13.1    12
```

Figure 3-10 Error Rate Per 1000 Delivered Source Instructions

The remaining screens pertain to option number five of the Model Menu, POLICY VARIABLES. Accept the preset values for the first seven screens by pressing <ENTER>, as prompted, for each screen. These preset values include: TASK UNDER-ESTIMATION=.35 (Figure 3-11), PERCENT OF EFFORT ASSUMED NEEDED FOR DEVELOPMENT=.85 (Figure 3-12), INITIAL

UNDERSTAFFING FACTOR=.4 (Figure 3-13), PERCENT OF EXPERIENCED EMPLOYEE EFFORT TO TRAIN A NEW EMPLOYEE=.5 (Figure 3-14), AVERAGE DAILY MANPOWER PER STAFF EXPENDED ON PROJECT TO TRAIN A NEW EMPLOYEE=.5 (Figure 3-15), FRACTION OF MANPOWER DEVOTED TO QUALITY ASSURANCE = .325,.29,.275, .255,.25,.275,.325,.375,.4,.4,0 (Figure 3-16), and WILLINGNESS TO CHANGE THE WORKFORCE = 0.0,.1,.4,.85,1,1,1, 1,1,1,1,1,1 (Figure 3-17).

```
                  SETTING POLICY VARIABLES

         1)  Press <ENTER> to accept the preset value
                              or
         2)  Enter the new value and press <ENTER>

The preset value of TASK UNDER-ESTIMATION = .35
```

Figure 3-11 Task Under-estimation

```
                  SETTING POLICY VARIABLES

         1)  Press <ENTER> to accept the preset value
                              or
         2)  Enter the new value and press <ENTER>

The preset value of PERCENT OF EFFORT ASSUMED NEEDED FOR

DEVELOPMENT = .85
```

Figure 3-12 Percent of Effort Assumed Needed for Development

```
                    SETTING POLICY VARIABLES

          1)  Press <ENTER> to accept the preset value
                               or
          2)  Enter the new value and press <ENTER>

The preset value of INITIAL UNDERSTAFFING FACTOR = .4
```

Figure 3-13 Initial Understaffing Factor

```
                    SETTING POLICY VARIABLES

          1)  Press <ENTER> to accept the preset value
                               or
          2)  Enter the new value and press <ENTER>

The preset value of PERCENT OF EXPERIENCED EMPLOYEE EFFORT

TO TRAIN A NEW EMPLOYEE = .25
```

Figure 3-14 Percent of Experienced Employee Effort to Train a New Employee

```
                    SETTING POLICY VARIABLES

          1)  Press <ENTER> to accept the preset value
                               or
          2)  Enter the new value and press <ENTER>

The preset value of AVERAGE DAILY MANPOWER PER STAFF

EXPENDED ON PROJECT TO TRAIN A NEW EMPLOYEE = .5
```

Figure 3-15 Average Daily Manpower Per Staff Expended on Project to Train a New Employee

```
                    SETTING POLICY VARIABLES

        1)  Press <ENTER> to accept the preset matrix
            values
                              or
        2)  Enter the new matrix values and press <ENTER>
            (Values must be separated by a space or comma)
            (To change any value in the matrix you re-enter
            all values)


The preset value of FRACTION OF MANPOWER DEVOTED TO QUALITY

ASSURANCE are:

1      2     3      4      5     6      7      8      9     10     11

.325   .29   .275   .255   .25   .275   .325   .375   .4    .4      0.
```

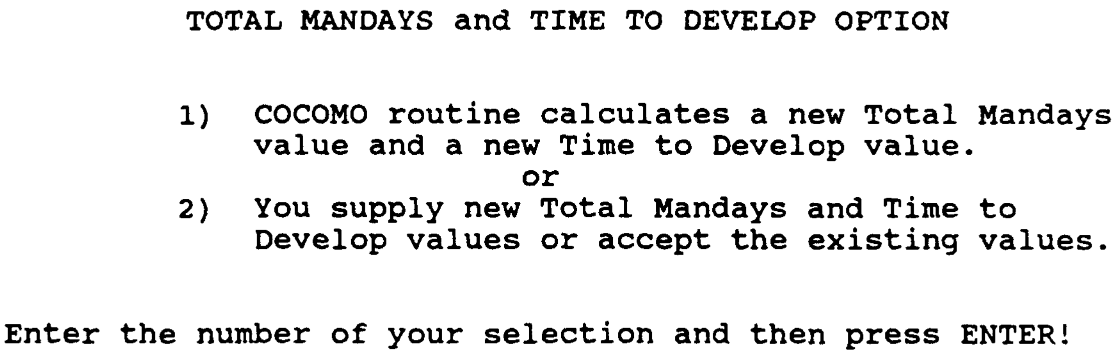
Figure 3-16 Fraction of Manpower Devoted to Quality Assurance

```
                    SETTING POLICY VARIABLES

        1)  Press <ENTER> to accept the preset matrix
            values
                              or
        2)  Enter the new matrix values and press <ENTER>
            (Values must be separated by a space or comma)
            (To change any value in the matrix you re-enter
            all values)


The old values of WILLINGNESS TO CHANGE THE WORKFORCE are:

  1    2    3    4    5     6    7    8    9    10   11   12   13   14

  0.   0.   .1   .4   .85   1.   1.   1.   1.   1.   1.   1.   1.   1.
```

Figure 3-17 Willingness to Change the Workforce

The next screen, Figure 3-18, prompts the user for a method to calculate TOTAL MANDAYS and TIME TO DEVELOP.

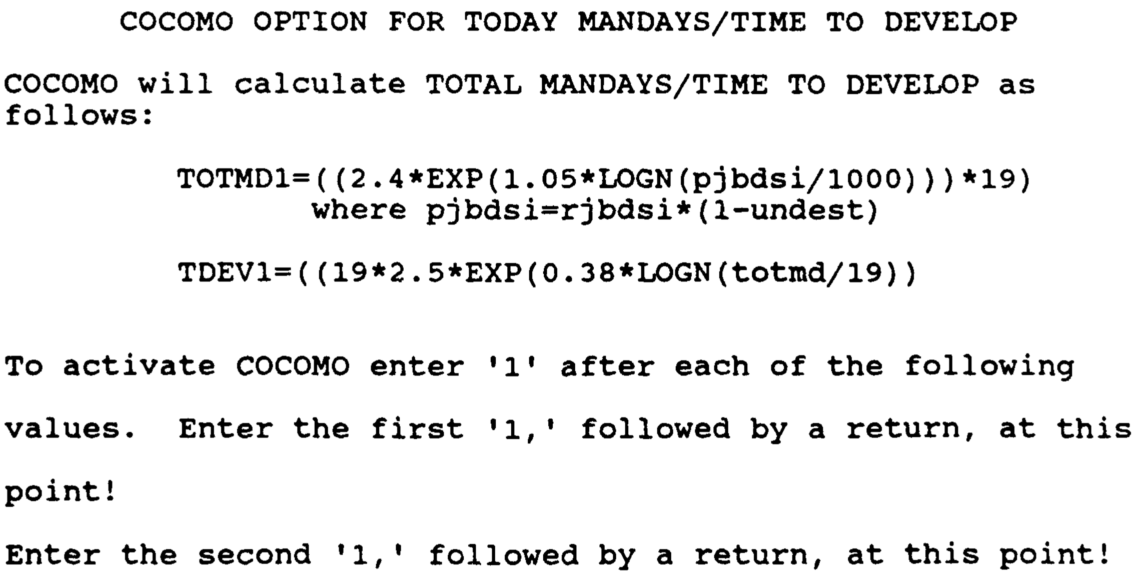
Choose option one--COCOMO. The COCOMO routine calculates a new TOTAL MANDAYS value and a new TIME TO DEVELOP value.



```
                 TOTAL MANDAYS and TIME TO DEVELOP OPTION

         1)  COCOMO routine calculates a new Total Mandays
             value and a new Time to Develop value.
                             or
         2)  You supply new Total Mandays and Time to
             Develop values or accept the existing values.

Enter the number of your selection and then press ENTER!
```

Figure 3-18 Total Mandays and Time to Develop Option

Next, Figure 3-19 prompts the user to enter a '1,' followed by a return, and then to enter a second '1,' followed by a return. This activates COCOMO.



```
      COCOMO OPTION FOR TODAY MANDAYS/TIME TO DEVELOP

COCOMO will calculate TOTAL MANDAYS/TIME TO DEVELOP as
follows:

         TOTMD1=((2.4*EXP(1.05*LOGN(pjbdsi/1000)))*19)
                where pjbdsi=rjbdsi*(1-undest)

         TDEV1=((19*2.5*EXP(0.38*LOGN(totmd/19))

To activate COCOMO enter '1' after each of the following
values.  Enter the first '1,' followed by a return, at this
point!
Enter the second '1,' followed by a return, at this point!
```

Figure 3-19 COCOMO Activation

This completes the walk-through of options 2,3,4,5 of the Model Menu. The next screen prompts the user to choose plot(s) 1,2,3,4. See Figure 3-20.

```
What output would you like to see?
    1.  PLOT1--SCHCDT,PJBSZ,JBSZMD,TOTWF,CUMMD
    2.  PLOT2--SCHCDT,PJBSZ,JBSZMD,CUMMD,TOTWF
    3.  PLOT3--CMTKDV,CUMTKT,CUMMD,PJBSZ,PDEVRC
    4.  PLOT4--AFMDPJ,JBSZMD,PJBSZ,PMDSHR

Pressing <ESC> after the plot(s) appear allows the user to
continue playing the game.
Pressing <QUIT> after the plot(s) appear finishes the Gaming
session and returns the user to the system prompt.
```

Figure 3-20 Plot Choice

Choose option 2, PLOT2. Note that pressing <ESC> after the plot(s) appear allows the user to continue playing the game. Pressing <QUIT> after the plot(s) appear finishes the Gaming session and returns the user to the system prompt. Press <2> followed by <ENTER> to display PLOT2 as in Figure 3-21. The variables plotted in PLOT2 are:

- SCHCDT--Estimated Schedule in Man Days
- PJBSZ--Perceived Project Size in Tasks
- JBSZMD--Estimated Project Cost in Man-Days

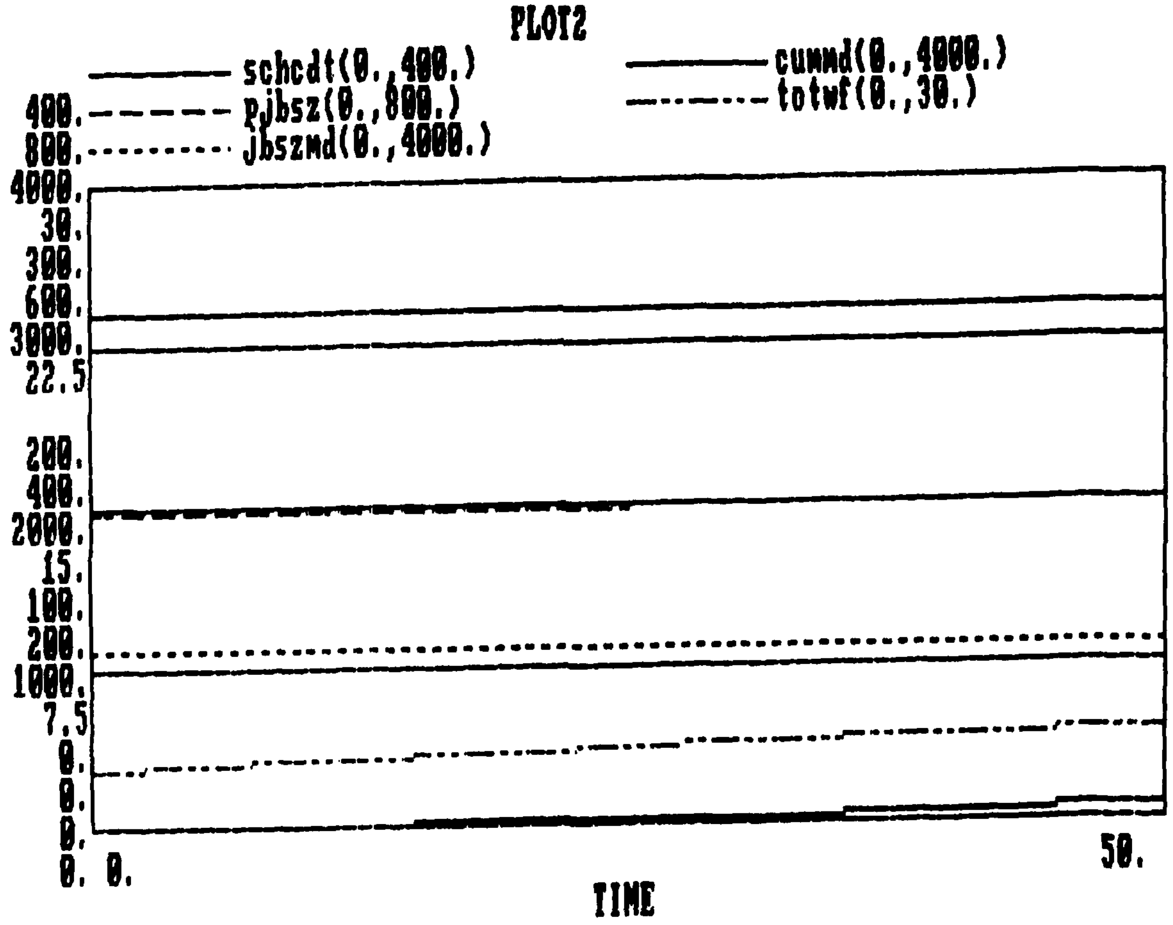
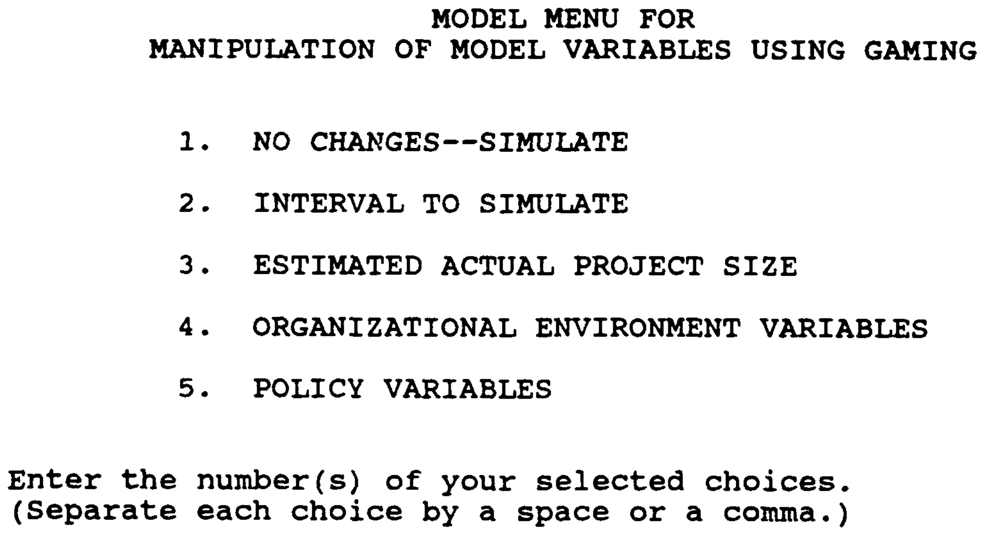


Figure 3-21 PLOT2, Time = 50

- CUMMD--Cumulative Man-Days Expended
- TOTWF--Total Workforce People.

The values depicted in Figure 3-21 for time interval 50 are: SCHCDT = 320, PJBSZ = 400, JBSZMD = 1100, CUMMD = 100, TOTWF = 4. After viewing PLOT2 press <ESC> to return to the Model Menu (Figure 3-22), to continue the Gaming session.



MODEL MENU FOR
MANIPULATION OF MODEL VARIABLES USING GAMING

1. NO CHANGES--SIMULATE
2. INTERVAL TO SIMULATE
3. ESTIMATED ACTUAL PROJECT SIZE
4. ORGANIZATIONAL ENVIRONMENT VARIABLES
5. POLICY VARIABLES

Enter the number(s) of your selected choices.
(Separate each choice by a space or a comma.)

Figure 3-22 Model Menu

All variables are now set to properly demonstrate Example One. Choose option 1, NO CHANGES--SIMULATE, to continue Gaming for the previously preset time interval of 50. Figure 3-20 again appears prompting the player to select a plot. Once again select PLOT2. Figure 3-23 is displayed showing time interval extended 50 more units to time interval 100. The values depicted in Figure 3-23 for time interval 100 are:

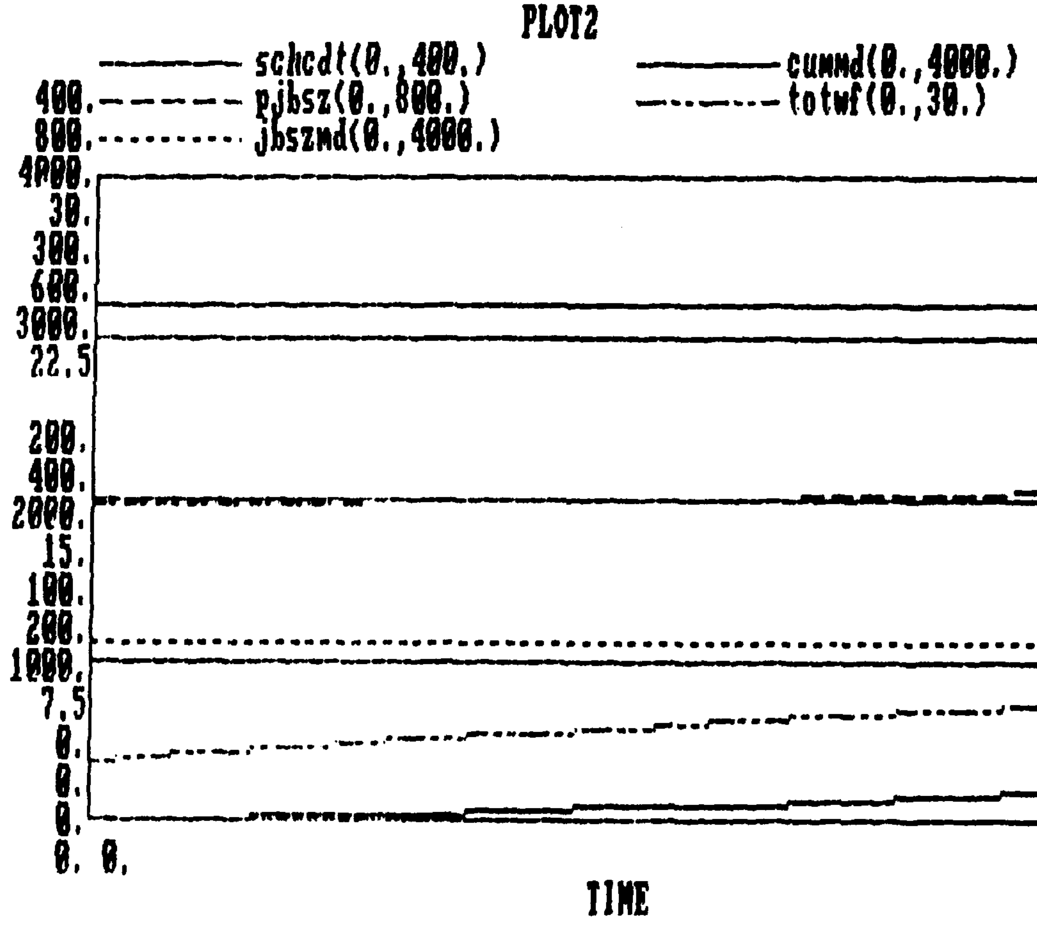


Figure 3-23 PLOT2, Time = 100

SCHCDT = 320, PJBSZ = 420, JBSZMD = 1100, CUMMD = 200, TOTWF = 6.

After viewing PLOT2 (Figure 3-23), press <ESC> to return to the Model Menu (Figure 3-22). Continue choosing option one of the Model Menu, NO CHANGES--SIMULATE, and option two of the plots, PLOT2, to continue viewing the simulation in additional increments of 50 time units until time = 400. Remember to stop after viewing Figure 3-29 with time = 400!!

The values in Table 3-1 are depicted in these series of plots.

TABLE 3-1

PLOT2 VALUES, TIME 0-400

Time	Figure	SCHCDT	PJBSZ	JBSZMD	CUMMD	TOTWF
50	3-21	320	400	1100	100	4
100	3-23	320	420	1100	200	6
150	3-24	320	450	1150	350	7
200	3-25	320	500	1250	550	8
250	3-26	320	570	1400	800	10
300	3-27	350	595	1450	1050	12
350	3-28	375	605	1900	1350	14
400	3-29	XXX	605	2050	1850	20.5

After viewing Figure 3-29 with time = 400, run the simulation one more time. During this simulation interval, the project reaches completion at time 420. The player has concluded the Gaming session by reaching the desired goal of project completion. Figure 3-30 depicts the following variable values at project completion:

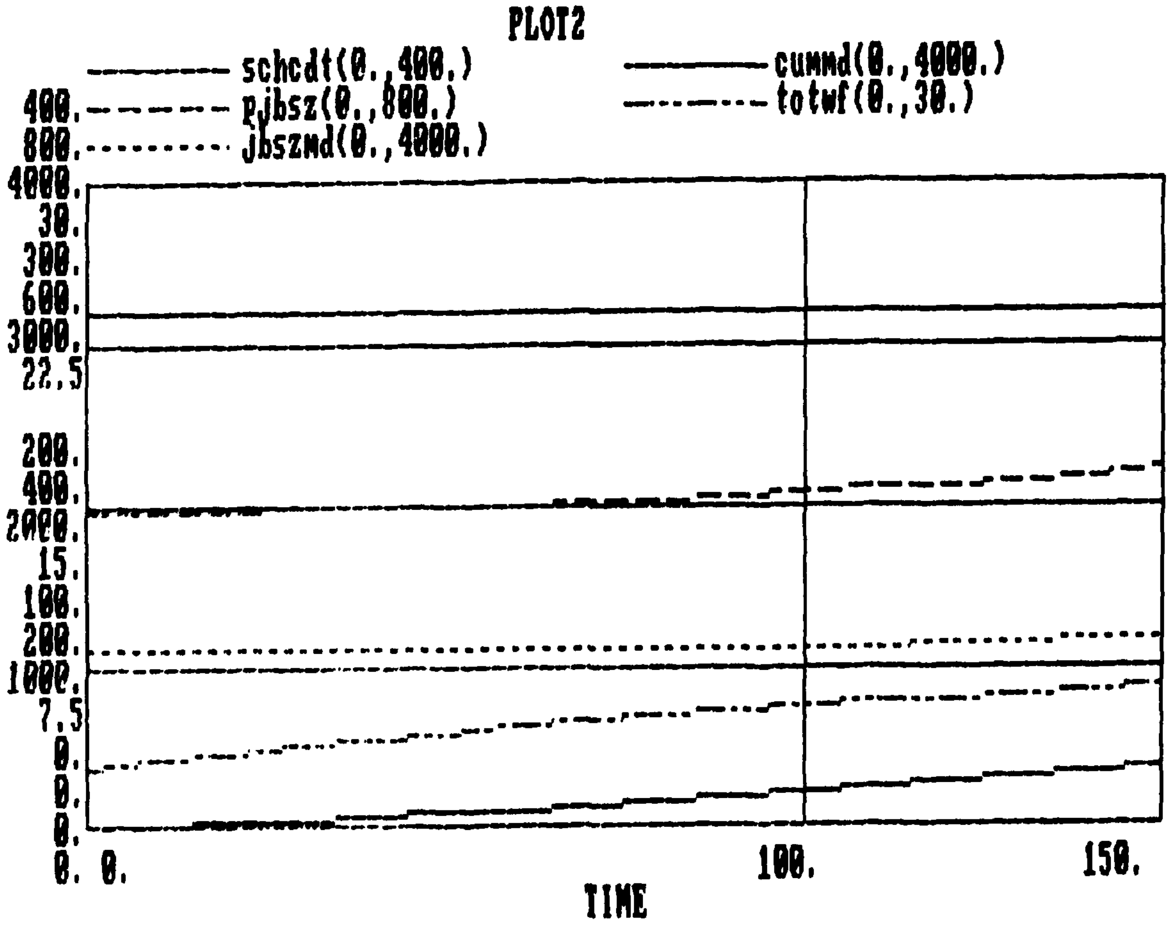


Figure 3-24 PLOT2, Time = 150

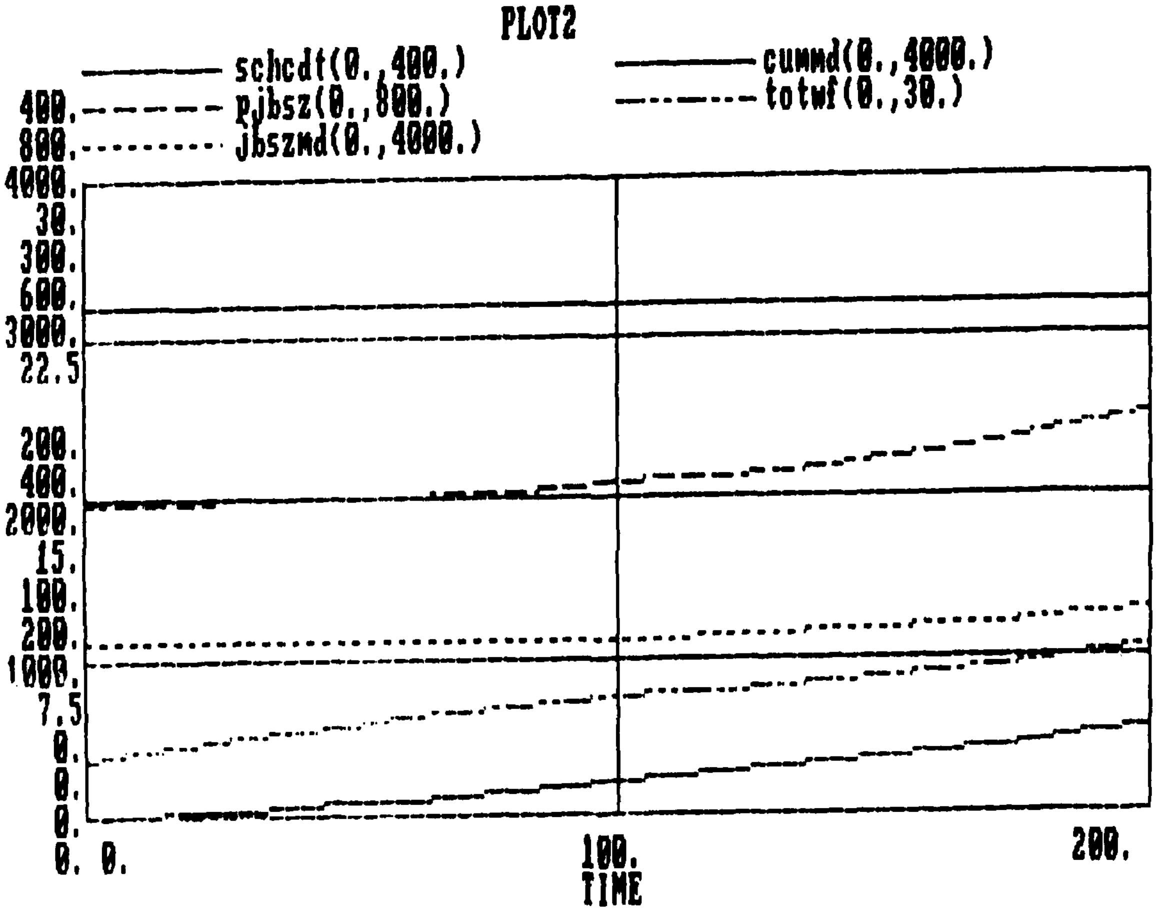


Figure 3-25 PLOT2, Time = 200

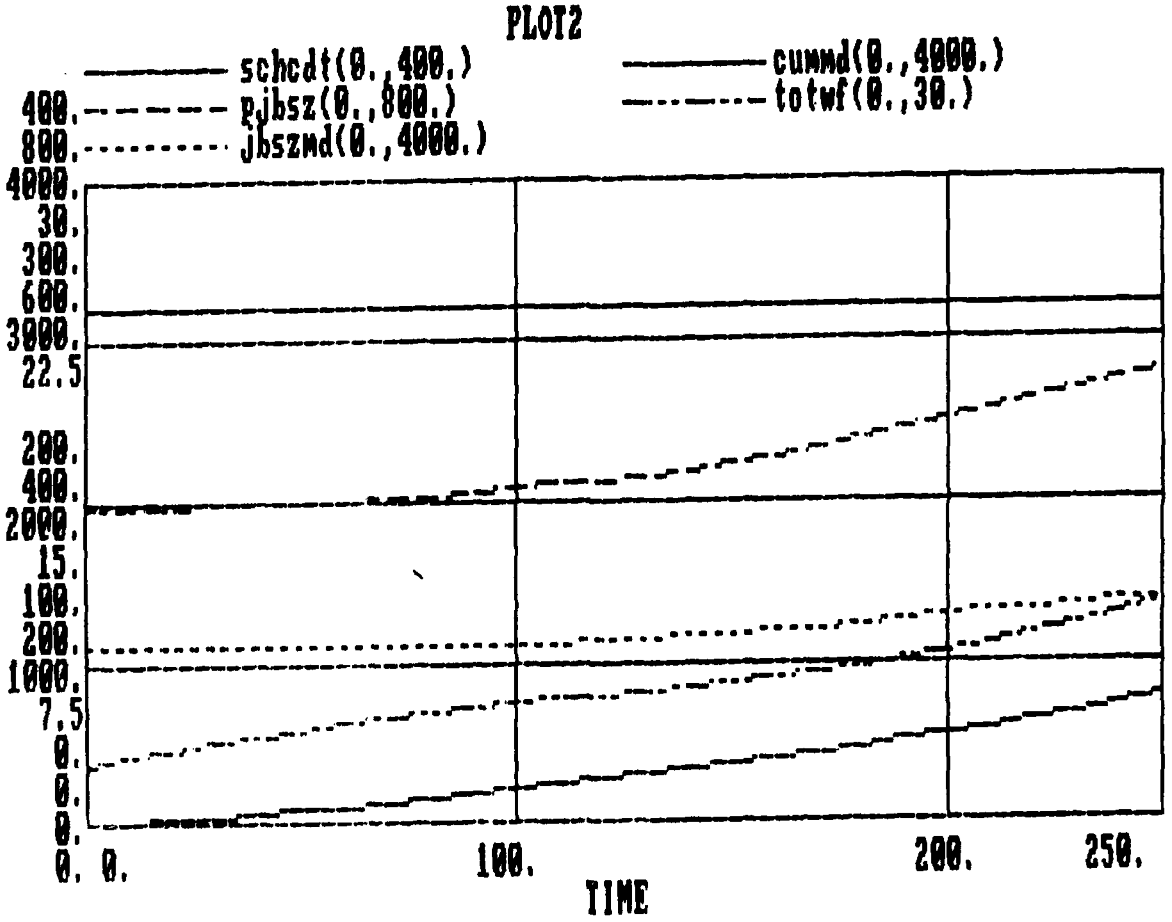


Figure 3-26 PLOT2, Time = 250

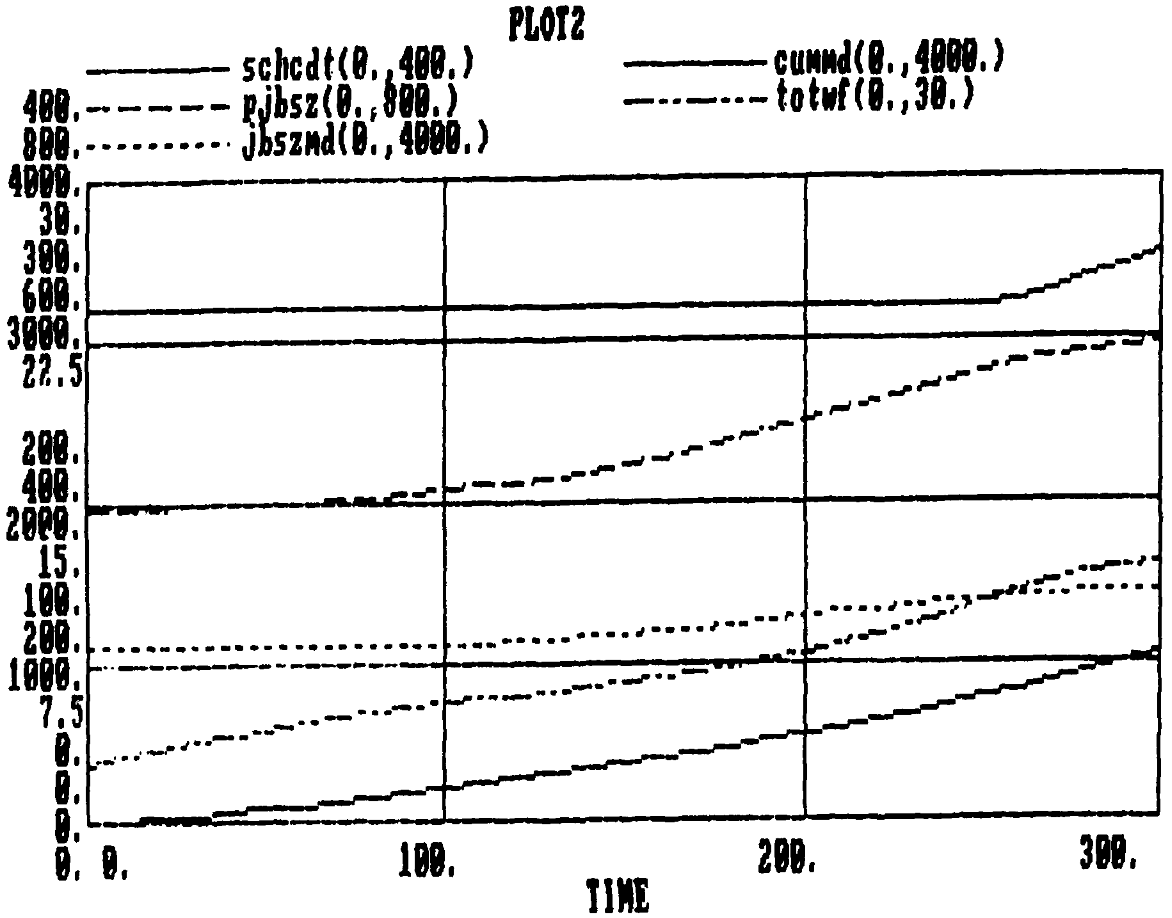


Figure 3-27 PLOT2, Time = 300

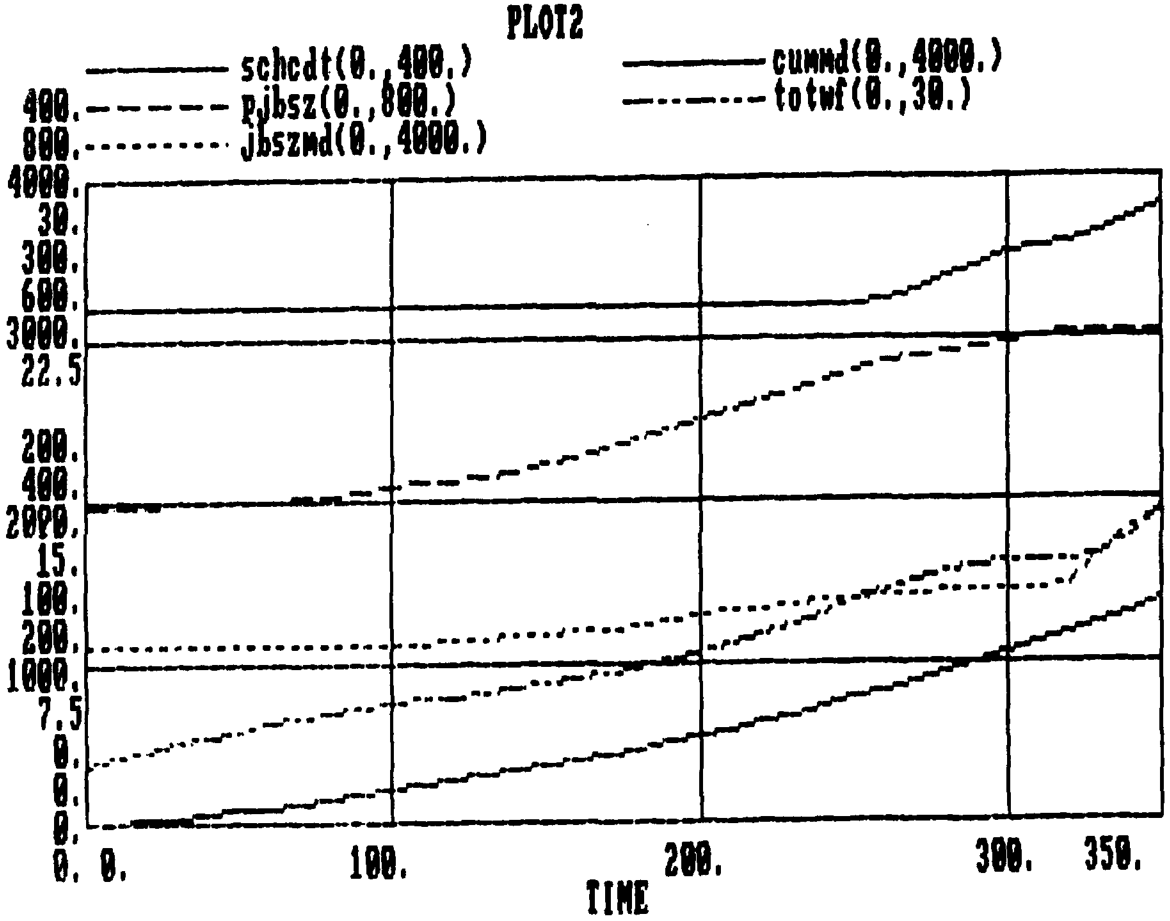


Figure 3-28 PLOT2, Time = 350

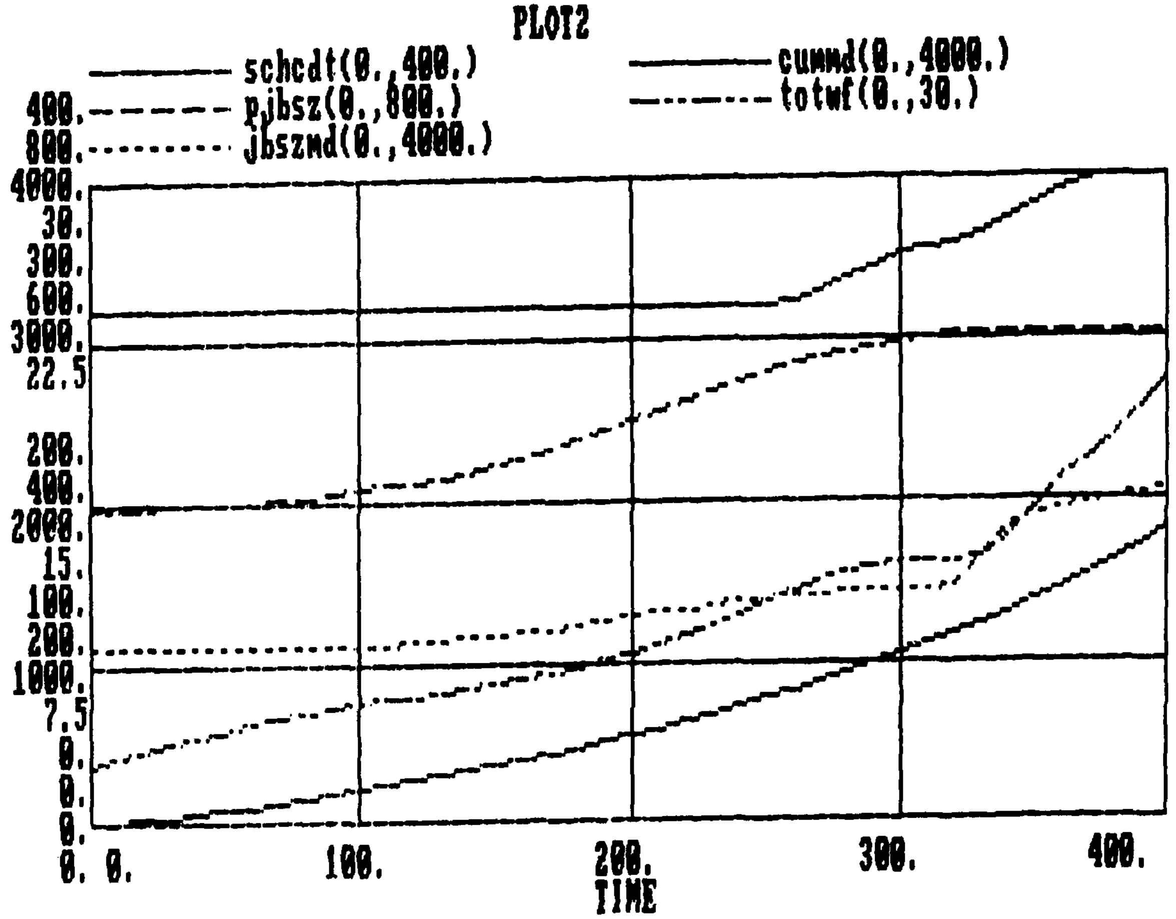


Figure 3-29 PLOT2, Time = 400

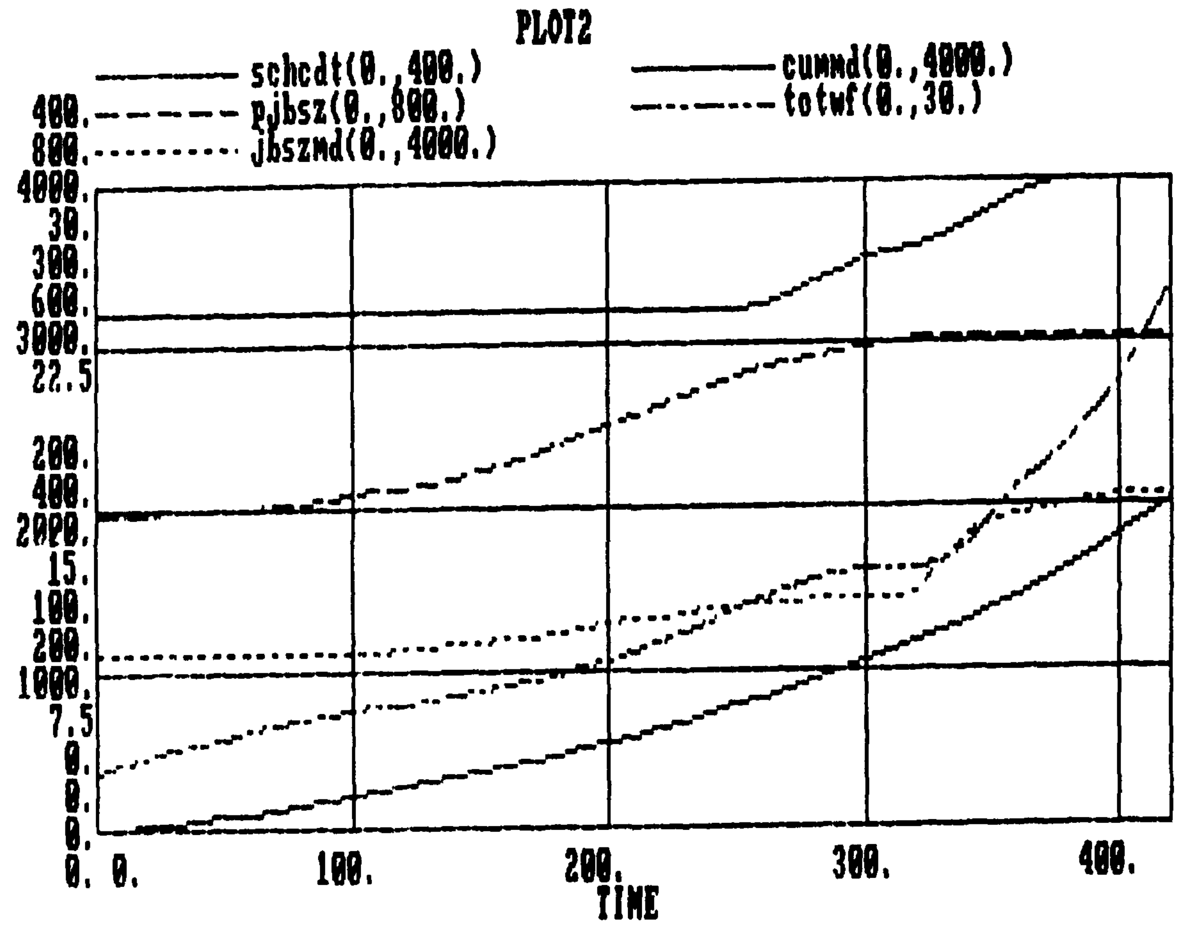


Figure 3-30 PLOT2, Time = 420

SCHCDT = XXX, PJBSZ = 605, JBSZMD = 2050, CUMMD = 2050, TOTWF = 24. These final results of Example One will be compared with the final results of Example Two to illustrate the effects of altering project variables.

B. EXAMPLE TWO

In Example Two, the player is also walked-through the same project as in Example One. However, at time intervals 50 and 200 the player will alter one project variable, HIRING DELAY (HIREDY). This helps illustrate the ability of Gaming to be used as a training tool by software project managers. It allows them to stop a simulation of a software development project at different intervals, assess project status, and react by altering project variables in real time.

Begin Example Two in the same manner as Example One. As a reminder, type <CMPL PROJECT> and press <ENTER>. After the model is compiled, type <GAME PROJECT> and press <ENTER>. Press <ENTER> to advance to page two of the Gaming Explanation and press <ENTER> once more to advance to the Model Menu (Figure 3-31).

```
                    MODEL MENU FOR
     MANIPULATION OF MODEL VARIABLES USING GAMING

          1.  NO CHANGES--SIMULATE

          2.  INTERVAL TO SIMULATE

          3.  ESTIMATED ACTUAL PROJECT SIZE

          4.  ORGANIZATIONAL ENVIRONMENT VARIABLES

          5.  POLICY VARIABLES

Enter the number(s) of your selected choices.
(Separate each choice by a space or a comma.)
```

Figure 3-31 Model Menu

All preset values in Example One will also be used in Example Two, except HIRING DELAY. Therefore choose option four of the Model Menu, ORGANIZATIONAL ENVIRONMENT VARIABLES. The first of five screens depicting ORGANIZATIONAL ENVIRONMENT VARIABLES appears in Figure 3-32. Press <ENTER> to accept the preset value of 40 for DELIVERED SOURCE INSTRUCTIONS PER TASK.

```
       SETTING ORGANIZATIONAL ENVIRONMENT VARIABLES

          1)  Press <ENTER> to accept the preset value
                             or
          2)  Enter the new value and press <ENTER>

The preset value of DELIVERED SOURCE INSTRUCTIONS PER TASK =
40
```

Figure 3-32 Delivered Source Instructions Per Task

Next, Figure 3-33 displays the preset value of HIRING DELAY = 30. Type <100> and press <ENTER> to change the variable HIRING DELAY from 30 to 100.

```
          SETTING ORGANIZATIONAL ENVIRONMENT VARIABLES

              1)  Press <ENTER> to accept the preset value
                                 or
              2)  Enter the new value and press <ENTER>

The preset value of HIRING DELAY = 30
```

Figure 3-33 Hiring Delay

Accept the preset values of ASSIMILATION DELAY = 20, AVERAGE EMPLOYMENT = 1000, and ERROR RATE PER 1000 DELIVERED SOURCE INSTRUCTIONS = 24, 22.9, 20.75, 15.25, 13.1, 12 by pressing <ENTER>, when prompted, for each screen as explained in Example One.

Figure 3-34 then prompts the user to select a plot(s). Choose option 2, PLOT2, as done in Example One.

```
What output would you like to see?
    1.  PLOT1--SCHCDT,PJBSZ,JBSZMD,TOTWF,CUMMD
    2.  PLOT2--SCHCDT,PJBSZ,JBSZMD,CUMMD,TOTWF
    3.  PLOT3--CMTKDV,CUMTKT,CUMMD,PJBSZ,PDEVRC
    4.  PLOT4--AFMDPJ,JBSZMD,PJBSZ,PMDSHR
```

Pressing <ESC> after the plot(s) appear allows the user to continue playing the game.

Pressing <QUIT> after the plot(s) appear finishes the Gaming session and returns the user to the system prompt.

Figure 3-34 Plot Choice

Figure 3-35 depicts the variable values for time interval 50 as: SCHCDT = 320, PJBSZ = 400, JBSZMD = 1100, CUMMD = 100, TOTWF = 4. Note that these values are identical to Figure 3-21 in Example One since all preset variables are equal.

After viewing Figure 3-35 press <ESC> to return to the Model Menu in Figure 3-36.

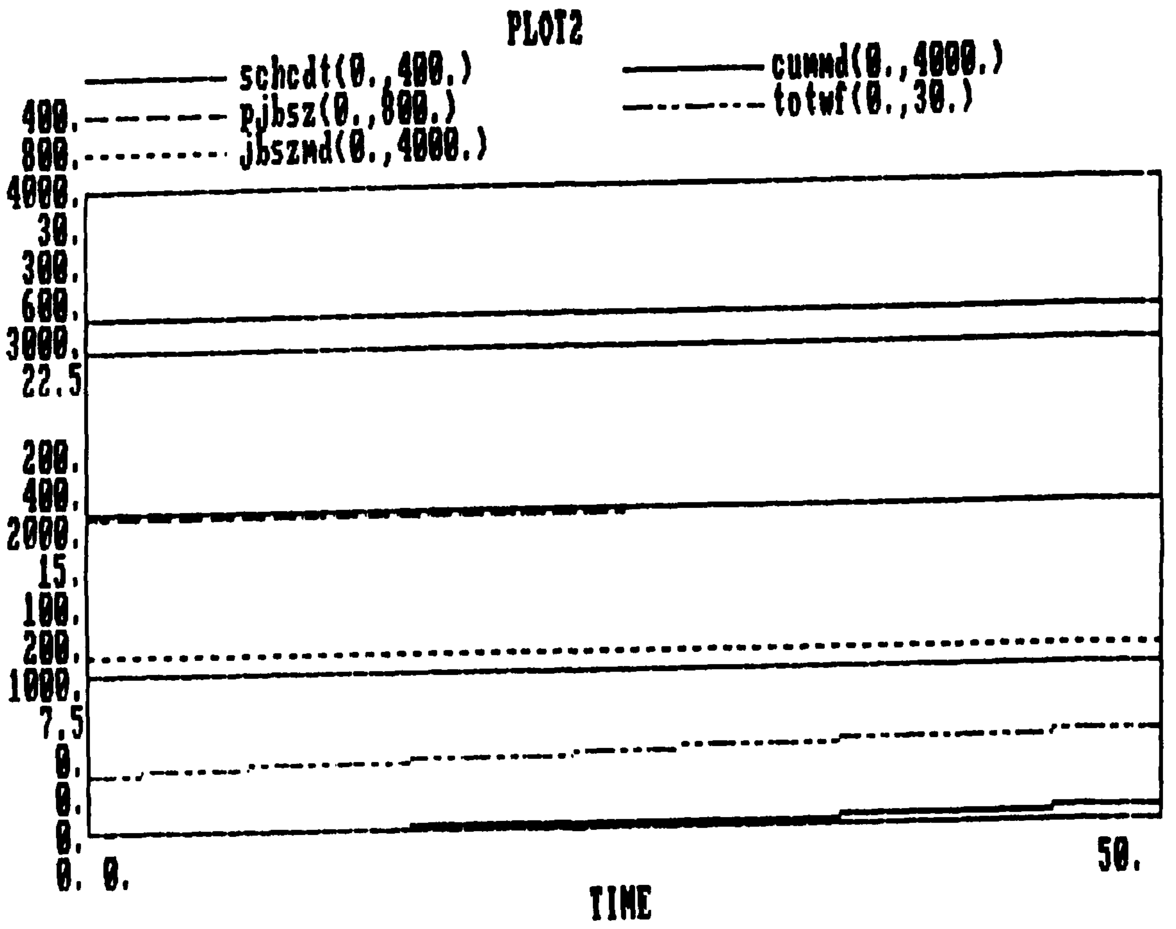


Figure 3-35 PLOT2, Time = 50

```
                  MODEL MENU FOR
     MANIPULATION OF MODEL VARIABLES USING GAMING

          1.  NO CHANGES--SIMULATE

          2.  INTERVAL TO SIMULATE

          3.  ESTIMATED ACTUAL PROJECT SIZE

          4.  ORGANIZATIONAL ENVIRONMENT VARIABLES

          5.  POLICY VARIABLES

Enter the number(s) of your selected choices.
(Separate each choice by a space or a comma.)
```

Figure 3-36 Model Menu

Once again select option 4, ORGANIZATIONAL ENVIRONMENT VARIABLES. Accept the preset value of 40 for DELIVERED SOURCE INSTRUCTIONS PER TASK.

Next, Figure 3-37 displays the preset value of HIRING DELAY = 100. This reflects the first variable change from 30 to 100. Type <50> and press <ENTER> to change the variable HIRING DELAY from 100 to 50.

```
       SETTING ORGANIZATIONAL ENVIRONMENT VARIABLES

          1)  Press <ENTER> to accept the preset value
                             or
          2)  Enter the new value and press <ENTER>

The preset value of HIRING DELAY = 100
```

Figure 3-37 Hiring Delay

Accept the preset values for the remaining ORGANIZATIONAL ENVIRONMENT VARIABLES. Select option two of plots, PLOT2, to view Figure 3-38 with time interval = 100. The variable values at time 100 are: SCHCDT = 320, PJBSZ = 420. JBSZMD = 1100, TOTWF = 7, CUMMD = 200.

Press <ESC> to return to the Model Menu. All preset values will remain the same until time = 200. Therefore, select option one of the Model Menu, NO CHANGES--SIMULATE, and option two of the plots, PLOT2, to continue Gaming for intervals of 50 until time = 200. Remember to pause after viewing Figure 3-40 where time = 200!! The values in Table 3-2 are depicted in these series of plots.

TABLE 3-2

PLOT2 VALUES, TIME 0-200

Time	Figure	SCHCDT	PJBSZ	JBSZMD	CUMMD	TOTWF
50	3-35	320	400	1100	100	4
100	3-38	320	420	1100	200	7
150	3-39	320	450	1200	400	7.5
200	3-40	320	510	1300	600	8.5

After viewing Figure 3-40 where time = 200, press <ESC> to return to the Model Menu. Choose option four, ORGANIZATIONAL ENVIRONMENT VARIABLES. Again, accept the preset value of 40 for DELIVERED SOURCE INSTRUCTIONS PER TASK.

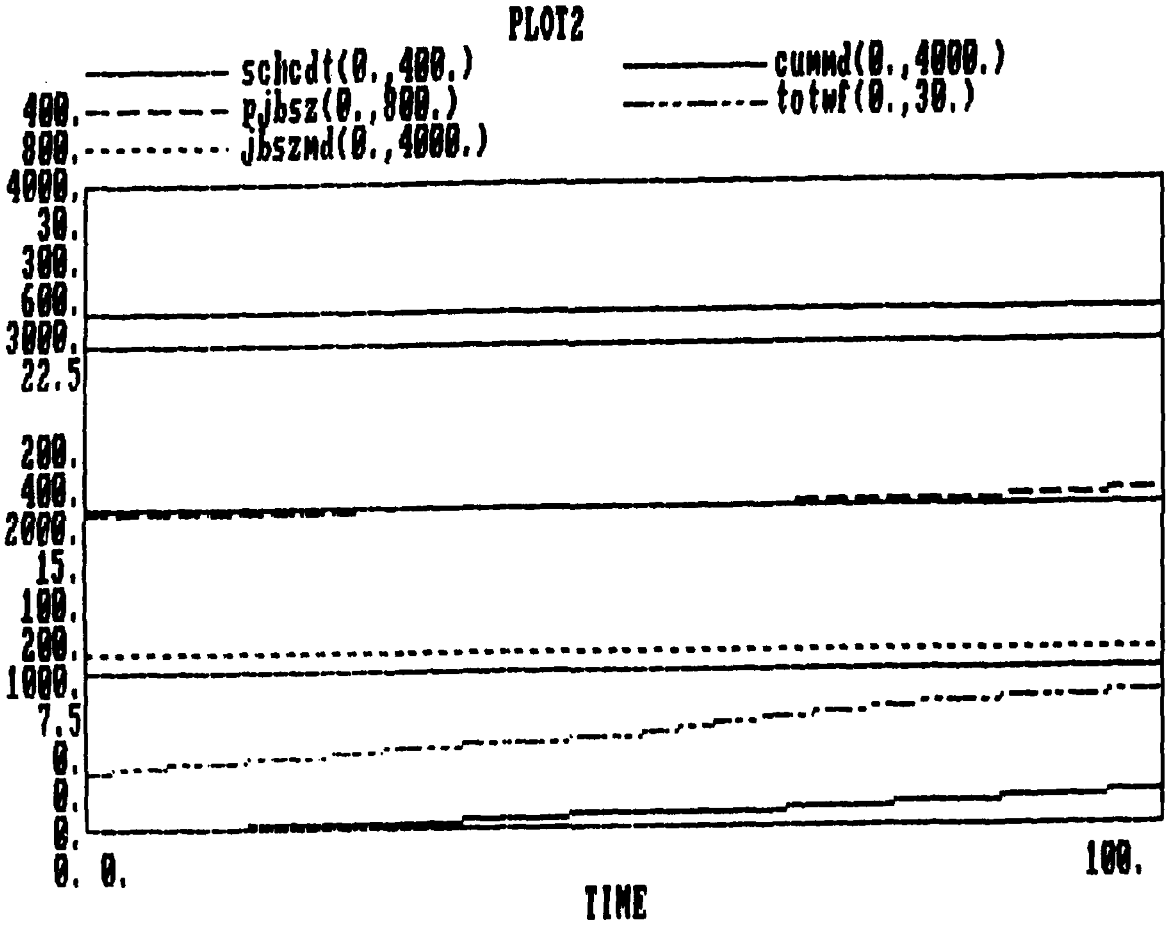


Figure 3-38 PLOT2, Time = 100

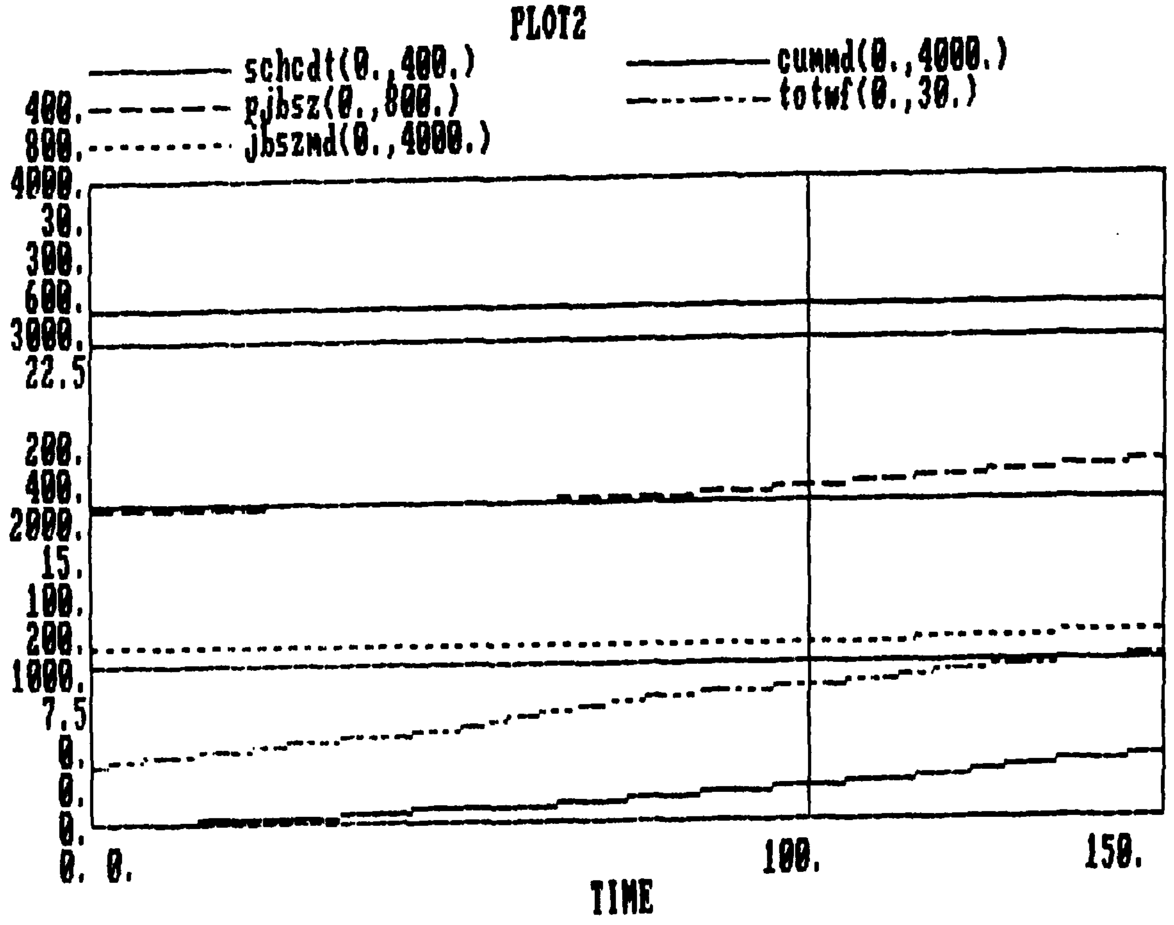


Figure 3-39 PLOT2, Time = 150

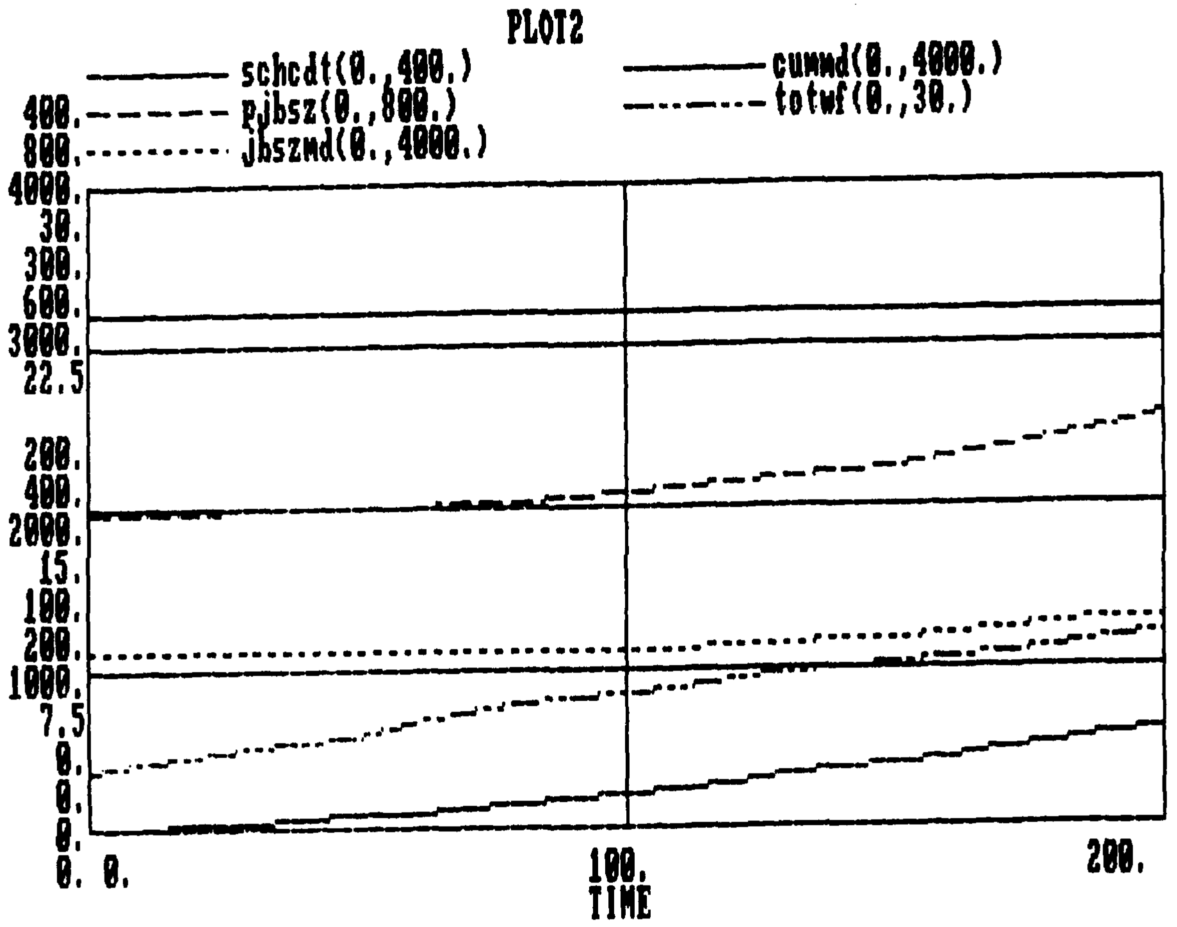
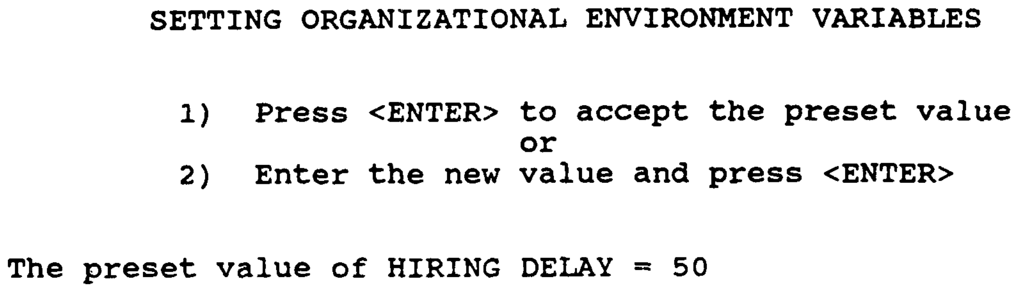


Figure 3-40 PLOT2, Time = 200

Next, Figure 3-41 displays the preset value of HIRING DELAY = 50. This reflects the second change of HIRING DELAY from 100 to 50. Type <10> and press <ENTER> to change the variable HIRING DELAY from 50 to 10.



SETTING ORGANIZATIONAL ENVIRONMENT VARIABLES

1) Press <ENTER> to accept the preset value
or
2) Enter the new value and press <ENTER>

The preset value of HIRING DELAY = 50

Figure 3-41 Hiring Delay

Accept the preset values for the remaining ORGANIZATIONAL ENVIRONMENT VARIABLES. Select option two of plots, PLOT2, to view Figure 3-42 where time = 250. The variable values at time 250 are: SCHCDT = 320, PJBSZ = 580, JBSZMD = 1400, TOTWF = 13.5, CUMMD = 900.

Press <ESC> to return to the Model Menu. Continue choosing option one of the Model Menu, NO CHANGES--SIMULATE, and option two of the plots, PLOT2, to continue viewing the simulation in additional increments of 50 time units until time = 350. The values in Table 3-3 are depicted in these series of plots.

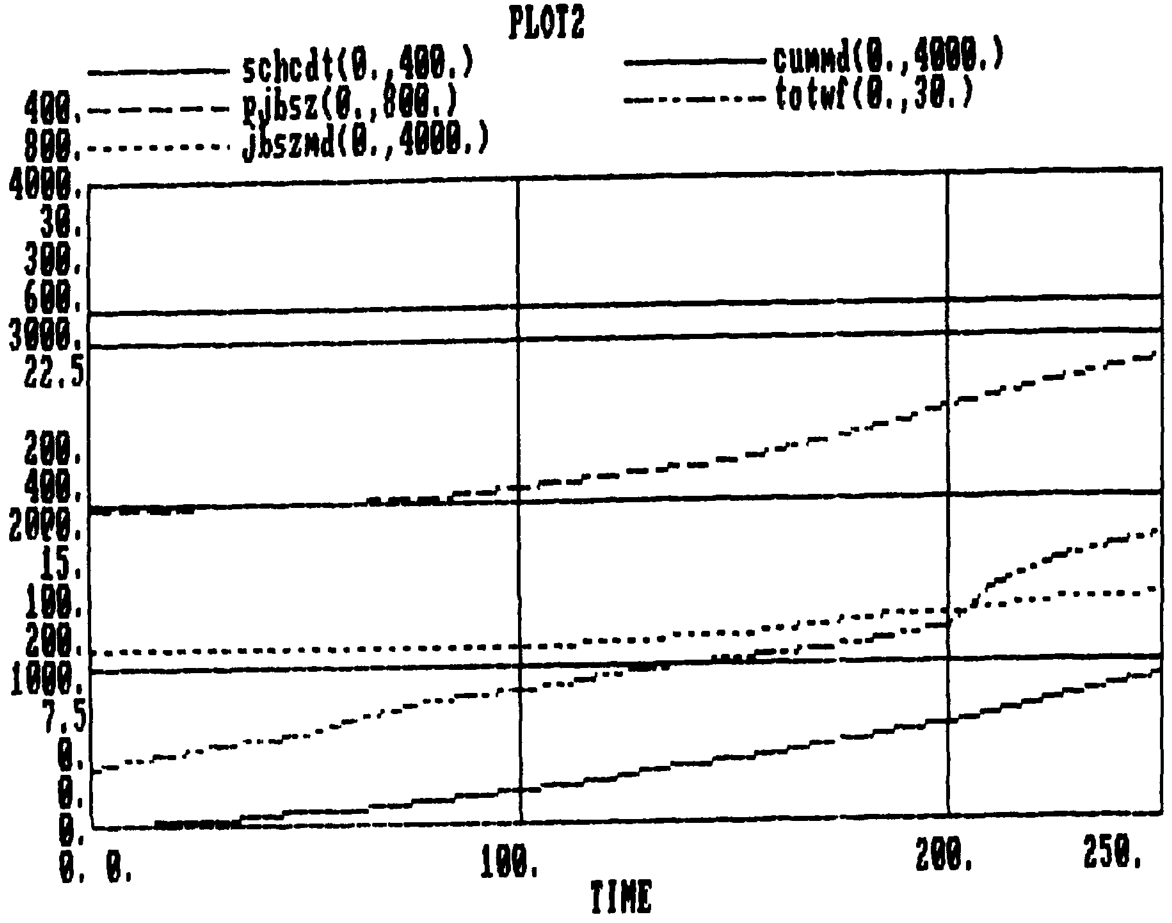


Figure 3-42 PLOT2, Time = 250

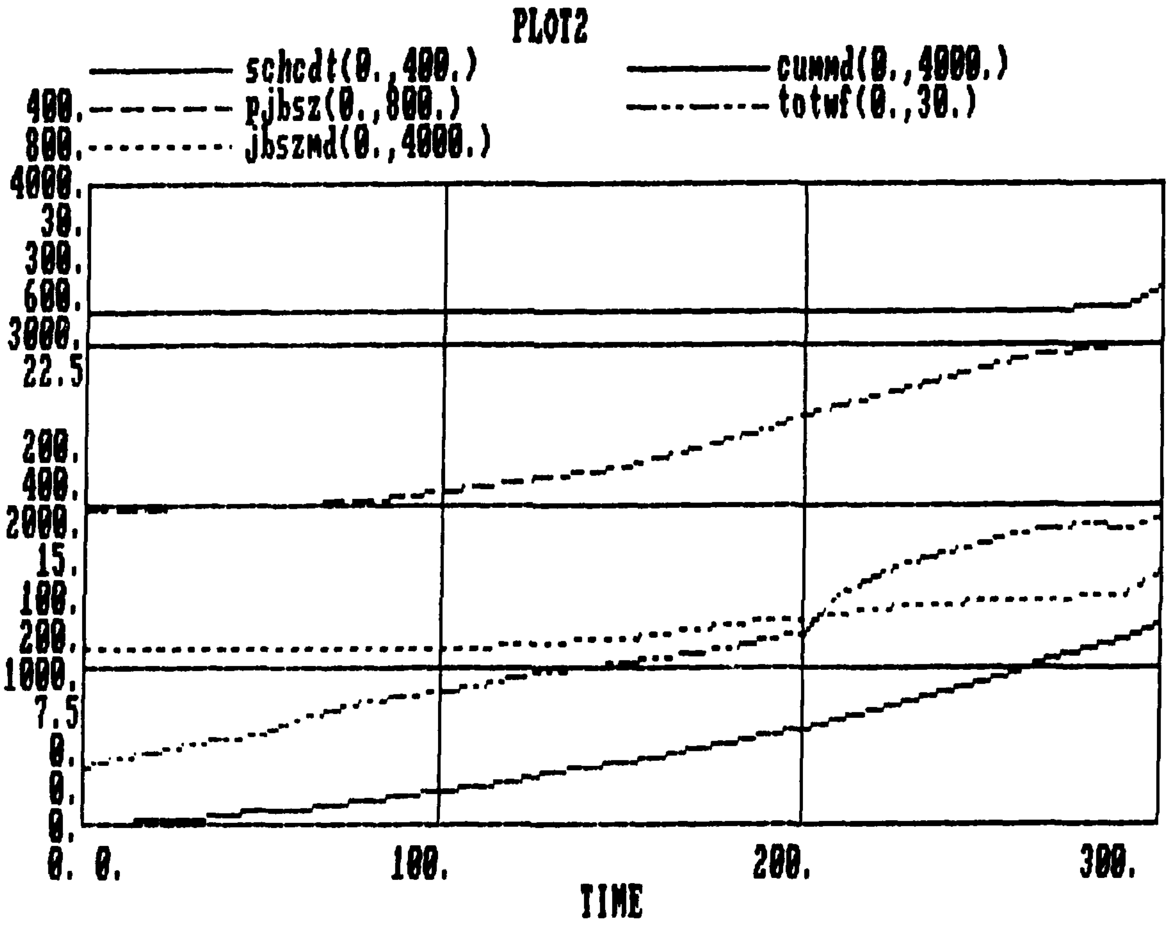


Figure 3-43 . PLOT2, Time = 300

TABLE 3-3

PLOT2 VALUES, TIME 200-350

Time	Figure	SCHCDT	PJBSZ	JBSZMD	CUMMD	TOTWF
200	3-40	320	510	1300	600	8.5
250	3-42	320	580	1400	900	13.5
300	3-43	335	600	1550	1200	14.5
350	3-44	365	605	2200	1300	XXX

After viewing Figure 3-44 where time = 350, run the simulation one more time. During this simulation interval, the project reaches completion at time = 380. The player has concluded the Gaming session by reaching the desired goal of project completion. Figure 3-45 depicts the following variable values at project completion: SCHCDT = 375, PJBSZ = 605, JBSZMD = 2950, CUMMD = 2950, TOTWF = XXX.

C. COMPARISON OF EXAMPLE ONE AND EXAMPLE TWO

Tables 3-4 and 3-5 are summaries of the final results for Example One and Example Two, respectively. Recall that all preset variables remain constant in both examples with the exception of HIRING DELAY (HIREDY). HIREDY is the average delay time, in work days, incurred in adding new staff members to the project. Example One held HIREDY = 100 for the duration of the simulation. In Example Two, HIREDY was changed at Time 50 to HIREDY = 50, and again at Time 200 to HIREDY = 10. HIREDY then remained equal to 10 for the duration of Example Two.

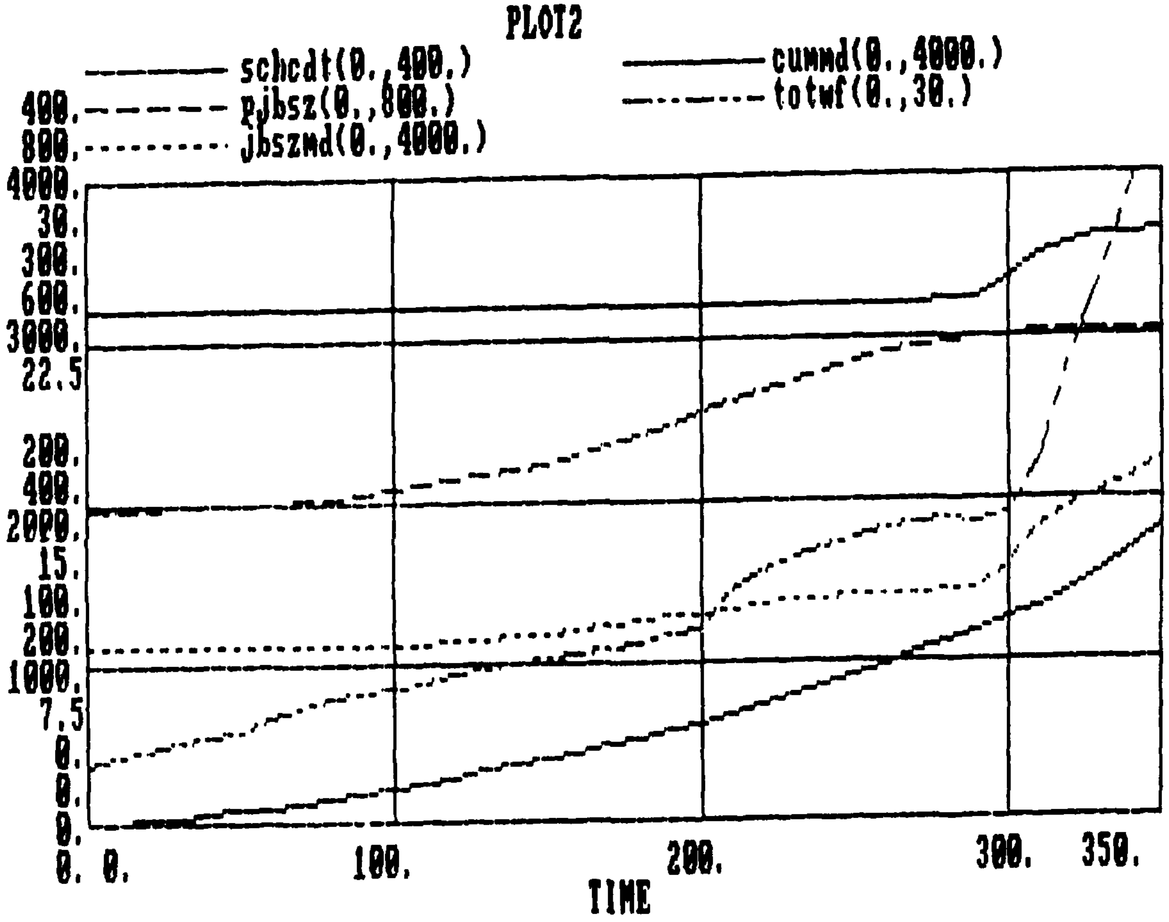


Figure 3-44 PLOT2, Time = 350

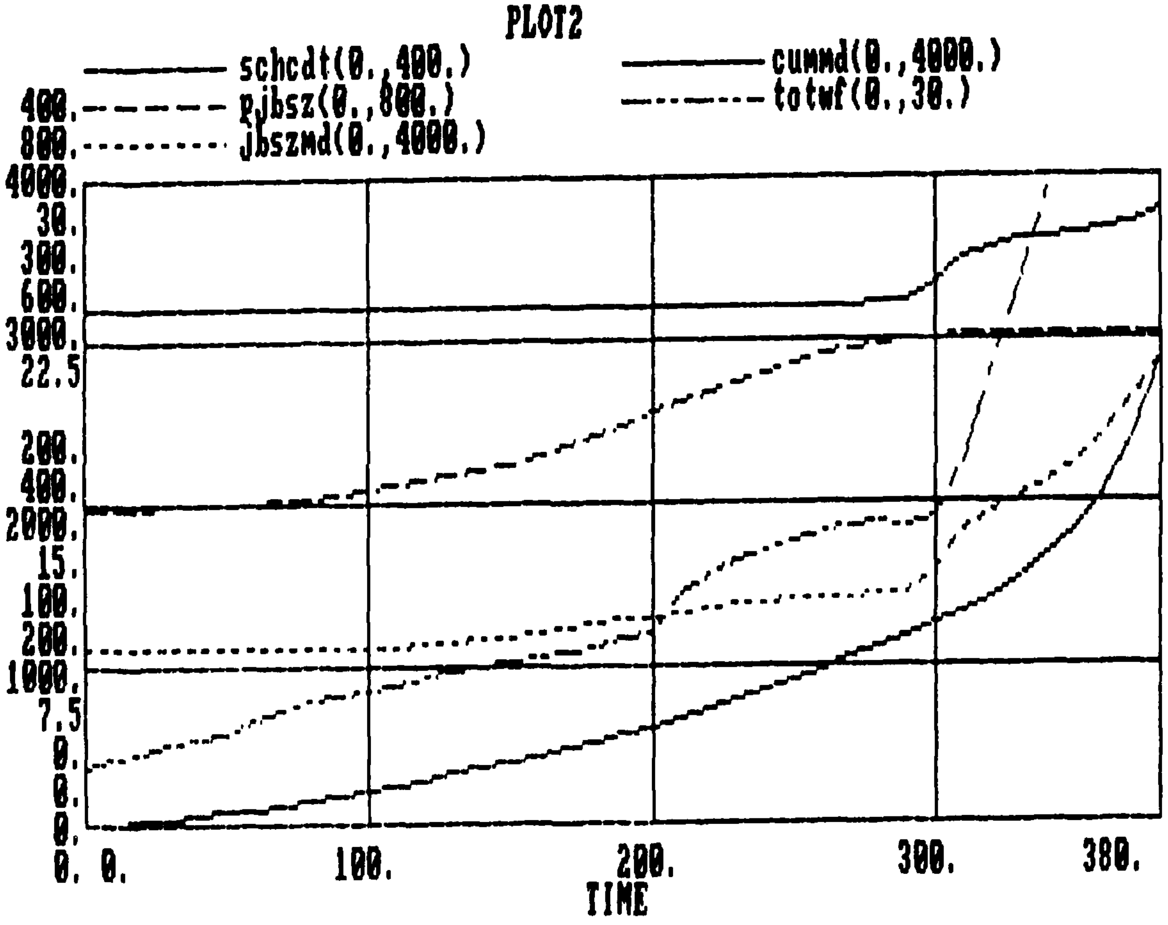


Figure 3-45 PLOT2, Time = 380

TABLE 3-4

EXAMPLE ONE RESULTS

Time	Figure	SCHCDT	PJBSZ	JMSZMD	CUMMD	TOTWF
* 50	3-21	320	400	1100	100	4
100	3-23	320	420	1100	200	6
150	3-24	320	450	1150	350	7
200	3-25	320	500	1250	550	8
250	3-26	320	570	1400	800	10
300	3-27	350	595	1450	1050	12
350	3-28	375	605	1900	1350	14
400	3-29	XXX	605	2050	1850	20.5
420	3-30	XXX	605	2050	2050	24

*HIREDY=100

TABLE 3-5

EXAMPLE TWO RESULTS

Time	Figure	SCHCDT	PJBSZ	JBSZMD	CUMMD	TOTWF
* 50	3-35	320	400	1100	100	4
** 100	3-38	320	420	1100	200	7
150	3-39	320	450	1200	400	7.5
***200	3-40	320	510	1300	600	8.5
250	3-42	320	580	1400	900	13.5
300	3-43	335	600	1550	1200	14.5
350	3-44	365	605	2200	1800	XXX
380	3-45	375	605	2950	2950	XXX

*HIREDY=100
**HIREDY=50
***HIREDY=10

D. VARIABLE ANALYSIS

1. SCHCDT--Estimated Schedule in Days

SCHCDT remains the same in both examples up through time = 250. At time = 300 in Example One, SCHCDT increases slightly more than it does in Example Two. This trend continues up through project completion where SCHCDT exceeds the boundary of the graph (greater than 400). This results from the fact that in Example Two, HIREDY is decreasing (100 to 50 to 10), enabling management to more rapidly increase the workforce, thus decreasing SCHCDT.

2. PJBSZ--Perceived Project Size in Tasks

PJBSZ remains nearly the same in both examples. Altering HIREDY has little effect on PJBSZ.

3. JBSZMD--Estimated Project Cost in Man-Days

JBSZMD begins to climb at a greater rate in Example Two at time 100 and continues to do so throughout project completion. This is in concurrence with the fact that as HIREDY decreases, TOTWF increases, thus increasing JBSZMD.

4. CUMMD--Cumulative Man-Days Expended and TOTWF--Total Workforce People

As the project nears completion in Example Two, CUMMD and TOTWF increase at a much greater rate than in Example One. Management chose to increase the workforce a lot towards the end of the project in Example Two by opting to decrease HIREDY from 100 to 50 and finally to 10. A smaller HIRING DELAY leads to a large influx of new people into the project, which in turn leads to a larger workforce size. This HIREDY reduction results in an earlier project completion time (time = 380), but at a much higher cost (CUMMD = 2950, TOTWF = OFF GRAPH).

IV. SYSTEM ARCHITECTURE

A. OVERVIEW OF THE SYSTEM ARCHITECTURE

A system architecture overview of the Gaming Facility created in this thesis is depicted in Figure 4-1. There are three interrelated subsystems represented at this level. They include: the Game Batch File, the Dynamica Model, and the Dynex Model Interface.

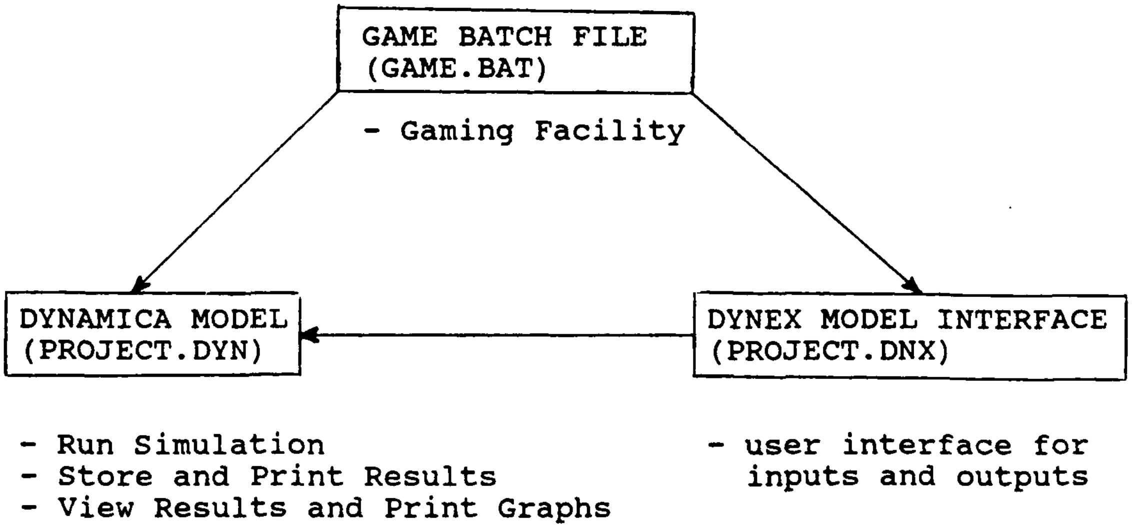


Figure 4-1. Overview of System Architecture

The heart of the system is the Dynamica Model. The Dynamica Model, written in Professional Dynamo, simulates a software project lifecycle. The Gaming Batch File, GAME. BAT is provided in the Professional Dynamo Software Package

[Ref. 9:pp. 1-10]. It enables the user to stop the simulation at pre-selected time intervals, view the status of the simulation, change variables, and continue the simulation. The GAME.BAT file also manages the Dynex Model Interface. The Dynex Model Interface allows the user to interact with the Dynamica Model by displaying a series of screens explaining Gaming and then prompting the player to type in values representing management decisions. PROJECT.DNX was created to specifically support Gaming. The Dynamica Model, using PROJECT.DYN, then simulates these management decisions for the time interval set by the player via Dynex. The Report module of Dynamica then displays the results from the beginning through the last simulation period in the form of four different plots. After viewing the plots, the player is again offered the opportunity to alter management decisions, continue the simulation, and view plots. This process continues until project completion.

B. GAME BATCH FILE

The batch file that initiates the Gaming Facility, GAME. BAT, is depicted in Figure 4-2.

Before executing the GAME.BAT file, the user must compile the original Dynamica Model, in this case PROJECT.DYN. To compile PROJECT.DYN the user need only type <CMPL PROJECT> and press <ENTER>. After the model is compiled, the GAME.BAT file may be executed by typing <GAME

```
 echo off
  if '%1' == '' goto expl
  smlt %1 -go = -prs = -ls
  if errorlevel 1 goto ext
:lp
  dynex %1 -in %1.stt -sc -ls
  if errorlevel 1 goto ext
  smlt %1 -gm = -ns
  if errorlevel 1 goto ext
  rep %1 -sc -ls
  if errorlevel 1 goto ext
  goto lp
:expl
  echo  This bat will manage DYNEX, SMLT, and REP to run a
  echo  game.  It assumes that the builder has created a
  echo  model, which has been compiled, and a dynex file to
  echo  guide the player.  The dynex file should contain rep
  echo  instructions that will be copied to model.drs.  It
  echo  is further assumed that the dynex file might display
  echo  variables from the model and, consequently, it must
  echo  be initially to produce a .STT file.
:ext
```

Figure 4-2 GAME.BAT Batch File

PROJECT> and pressing <ENTER>. GAME.BAT begins by simulating the Dynamica Model, PROJECT.DYN, for a very short period of time (1e-30). This 'tricks' simulate into stopping before the first DT, thus initializing the model. Once the model is initialized, it is preserved in an .STT file and later used by Dynex. Preserving a model in an .STT file enables the user to continue Gameplay from the finishing time of the previous simulation. The GAME.BAT file then manages the three Professional Dynamo Modules: Dynex, Simulate, and Report to complete the Gaming Facility.

C. THE DYNAMICA MODEL

The Dynamica Model is written in Professional Dynamo [Ref. 9]. The name of the program that supports Gaming is PROJECT.DYN. All interaction between the user and PROJECT.DYN is managed transparently by GAME.BAT or the Dynex Model Interface.

The following code appears in PROJECT.DYN to specifically support the Gaming Facility:

- SAVPER=1. SAVPER is the interval of time between the saving of simulation results for later comparative output.

- MD_INTERVAL=50. MD_INTERVAL is the manager's decision on how long to run a simulation before stopping to view its status, change variables, and then continue the simulation. Fifty is the default value, but the user has the option of changing this variable throughout the duration of Gameplay.

- MD_LENGTH=1e-30. Setting MD_LENGTH to a small value (1e-30) 'tricks' the Simulator into initializing the model, but stopping before the first DT. Once the model is initialized, it can be preserved in a .STT file. The .STT file is the file where the Simulator stores the final variable values from the run that you are preserving. Preserving a file allows a user to preserve model values so that simulation can be resumed with those conditions.

- TMSTOP.k=CLIP(TIME.k,1000,PTKTST.k,.99). The CLIP statement assigns a value to variable TMSTOP. The value assigned is equal to the lesser value of TIME.k (with a max value of 1000) or whenever PTKTST (percentage of tasks tested) reaches 99% completion.

- LENGTH.k=MIN(MD_LENGTH,TMSTOP.k). LENGTH is the value of time at which the simulation is to be terminated. The MIN statement assigns a value to variable LENGTH. LENGTH will be set equal to the lesser of the two values MD_LENGTH or TMSTOP.

- TM.k=TIME.k. Adding the extra variable TM set to the value of TIME allows the player to plot variables against this extra TIME with an XY plot.

- SAVE TM. The SAVE statement is used for selecting variable values to be saved, in this case variable TM.

D. DYNEX INTERFACE

The Dynex Model Interface for this thesis was written in Dynex (DYNAMO for Executives). Dynex is a model interface that allows a user with no knowledge of Professional Dynamo to simulate a model and view the results. Using Dynex, the experienced model builder can make a model available for use in a structured and easily understood framework. By responding to simple questions and prompts, the game player can alter project variables, execute simulations, and view results of those simulations.

The following code appears in PROJECT.DNX to specifically support the Gaming Facility:

```
- if #tm<.1 then
     [text - Gaming Explanation]
  else
  end
```

This coding allows the Gaming Explanation to be suppressed after initial viewing. At the beginning of Gaming, tm is initialized to zero, making tm<.1 a true statement, thus displaying the Gaming Explanation. After the initial run of a simulation, tm will be greater than .1, thus bypassing the Gaming Explanation and moving the user directly to the Model Menu shown in Figure 4-3. Menu options are explained in Chapter III.

- dq md_interval=50. This 'dq' or 'decision query' statement controls the manager's decision on how long to run a simulation before stopping to view its status, possibly change variables, and then continue the simulation. Fifty is the default value, but the user has the option of changing this variable throughout the duration of Gameplay.

- SPEC MD_LENGTH=#LENGTH+MD_INTERVAL. The SPEC statement obligatorily increases MD_LENGTH so that Simulate will stop at the appropriate time as defined in the previously discussed PROJECT.DYN coding. #LENGTH directs DYNEX to make the calculation using the length value found in the .STT file.

```
                      MODEL MENU FOR
       MANIPULATION OF MODEL VARIABLES USING GAMING

           1.  NO CHANGES--SIMULATE

           2.  INTERVAL TO SIMULATE

           3.  ESTIMATED ACTUAL PROJECT SIZE

           4.  ORGANIZATIONAL ENVIRONMENT VARIABLES

           5.  POLICY VARIABLES

Enter the number(s) of your selected choices.
(Separate each choice by a space or a comma.)
```

Figure 4-3 Model Menu

V. CONCLUSIONS

A. ACCOMPLISHMENTS

The primary objective of this thesis was to enhance the user interface to the Dynamica Model of Software Project Management by incorporating Gaming. Gaming allows software project managers to stop a simulation of a software development project at different intervals, assess project status, and react by altering project variables in real time. Secondly, this thesis illustrated the effects of dynamic decision making, using Gaming, through the comparison of two software project development examples.

B. LESSONS LEARNED

The design of the Gaming interface is best understood by first analyzing an overview of the system architecture. The overview included breaking down the system architecture into its three interrelated subsystems: the Game Batch File, the Dynamica Model, and the Dynex Model Interface. Once these three interrelated subsystems are sufficiently understood, the user can then effectively isolate the interrelationships of key variables to successfully design the Gaming interface.

C. FUTURE DIRECTION

The current Gaming interface for the Dynamica Model of Software Project Management could be expanded in the following ways:

- Allow users to enter qualitative values for managerial decisions. Ask whether the player wants a "large" or "small" effect, i.e., using "Fuzzy Set Theory" ideas.
- Provide on-line help facilities within the Dynex Model Interface. Dynex allows the builder to provide additional detail about any query or choice by writing a help file for it.
- Allow a user to save a Gaming session to a file for future reference.
- Provide a summary table of variable values for each simulation interval. This table would list all previous managerial decisions during the simulation for quick user reference.

Finally, the current Gaming interface could be tested and evaluated by users to determine difficulties with the design. These results would be documented and studied to further improve the main goal of the Dynamica Model of Software Project Management; to aid the software project manager in understanding the software development process.

LIST OF REFERENCES

1. Musa, J.D., "Software Engineering: The Future of a Profession," IEEE Software, January 1985.

2. Buckley, F. and Poston, R., "Software Quality Assurance," IEEE Software Engineering, January 1984.

3. Newport, John P. Jr., "A Growing Gap in Software," Fortune, April 1986.

4. Abdel-Hamid, Tarek K. and Madnick, Stuart E., "Managing Software Projects: An Integrative System Dynamics Modeling Approach," Naval Postgraduate School Working Paper No. 87-14, Monterey, California, June 1987.

5. Haury, Carson E., Design and Development of a User Interface for the Dynamica Model of Software Project Management, Master's Thesis, Naval Postgraduate School, Monterey, California, March 1988.

6. Thayer, R.H., et al., "Major Issues in Software Engineering Project Management," IEEE Transactions on Software Engineering, Vol. SE-7, No. 4, July 1981.

7. Abdel-Hamid, Tarek K. and Madnick, Stuart E., "An Integrative System Dynamics Perspective of Software Project Management: Arguments for an Alternative Research Paradigm," Naval Postgraduate School Working Paper No. 87-09, Monterey, California, May 1987.

8. Richardson, G.P., and Pugh, G.L., III, Introduction to System Dynamics Modeling with Dynamo, Cambridge, Massachusetts, 1981.

9. Pugh-Roberts Associates, Inc., "Trainer Framework," Professional Dynamo Plus Reference Manual, 1986.

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Technical Information Center 2
Cameron Station
Alexandria, Virginia 22304-6145

2. Library, Code 0142 2
Naval Postgraduate School
Monterey, California 93943-5002

3. Dr. Tarek K. Abdel-Hamid, Code 54Ha 10
Department of Administrative Sciences
Naval Postgraduate School
Monterey, California 93943-5000

4. Prof. Magdi Kamel, Code 54Ka 2
Department of Administrative Sciences
Naval Postgraduate School
Monterey, California 93943-5000

5. Edward Tulenko 2
140 East Fourth Avenue
Latrobe, Pennsylvania 15650

6. Computer Technology Curricular Office, Code 37 1
Naval Postgraduate School
Monterey, California 93943-5000

7. COMMANDANT (G-PTE-1) 1
U.S. Coast Guard
2100 Second Avenue
S.W. Washington, D.C. 20593-0001

8. Major Paul W. LeBlanc 1
MCCDPA
Marine Corps Combat Development Command
Quantico, Virginia 22134-5010

9. Chief of Naval Operations, OP-945 1
Director, Information Systems
Washington, D.C. 20350-2000