Third International Working Conference

on

Dynamic Modelling of Information Systems

June 9-10, 1992

Noordwijkerhout, the Netherlands

Participants Edition

Proceedings of the "Third International Working
Conference on Dynamic Modelling of Information Systems"
Noordwijkerhout, The Netherlands, June 9-10, 1992
TU Delft, Department of Information Systems, 1992

THE DECISION-SUPPORT EFFECTIVENESS OF A SIMULATION ENVIRONMENT FOR INFORMATION SYSTEMS ANALYSTS: AN EXPLORATORY STUDY

James R. Warren, A. F. Norcio, Jack W. Stott, G. C. Canfield, and R. W. Freedman
Department of Information Systems
University of Maryland Baltimore County
Baltimore, MD 21228
(410) 455-3206
norcio@afn.ifsm.umbc.edu

KEYWORDS: Simulation environment, decision support, CASE .

ABSTRACT

This paper describes an exploratory study to assess the effect of automated support on the accuracy with which information systems (IS) analysts answer questions about the dynamics of IS designs. The term *CASE/simulation system* designates an architecture allowing the automatic production of system simulations of IS designs from the data in a CASE-tool (computer-aided software engineering tool) data dictionary without the writing of computer simulation programs by the IS analyst. A prototype CASE/simulation system has been developed which provides intelligent support in the interpretation of simulation output. Pilot study results indicate that the prototype improves the accuracy with which analysts answer questions for which responses are mathematically difficult to formulate, and that availability of simulation results improves confidence in responses. The implications of the findings for CASE/simulation systems and future research directions are discussed.

1. INTRODUCTION

IS developers frequently benchmark the performance of the hardware and software components of proposed IS designs, but modelling the overall dynamics of the IS design is not a normal part of the system development process. We consider the analysis of overall IS design dynamics to entail modelling the extent to which a set of system components in a design (computer-based systems, people, and non-information-processing machinery interacting with the IS) satisfies its performance requirements. It is unfortunate that this variety of analysis is neglected, because IS dynamics are an important consideration in many design decisions.

Concerns about performance shape the choices of hardware and database architecture. IS developers must be sure that processing efficiency is adequate to achieve the dynamic requirements of the overall system (e.g., time to fill a customer order). Without a comprehensive model of IS dynamics, however, these critical choices must be highly conservative, and based more on experience with similar designs than on analysis. Such blindly conservative decision-making impedes the use of newer technologies. Largely unfounded performance concerns hampered the adoption of the relational DBMS model [6] and are a current barrier to newer data models such as the semantic [11, 14] and functional data models [18, 4, 5]. Efforts at hardware down-sizing are also impaired by the threat of inadequate performance. Furthermore, determining which processes within a system are truly in need of automation is a fundamental decision, based largely on the criterion of adequate dynamic performance, which deserves the careful attention due all requirements-oriented issues.

The objective is to make analysis of overall IS design dynamics a routine part of IS development. To achieve this objective, the cost of dynamic analysis must be lowered; that is, access to dynamic analysis of designs must be made more convenient to IS analysts. The term *CASE/simulation system* designates an architecture allowing the automatic production of system simulations of IS designs from the data in a CASE-tool (computer-aided software engineering tool) data dictionary without the writing of computer simulation programs by the IS analyst. A CASE/simulation system provides convenient performance analysis via system simulation.

1.1. Background

The literature suggests several trends relevant to CASE/simulation systems:

1. The delivery of performance modelling capability to IS analysts with models tailored to the specific needs of IS dynamics modelling. For example, QASE RT by Advanced Systems Technologies [8] supports modelling of wide-area and local-area networks, where the application code size, type of hardware, and operating system of each computer in the network is considered. A prototype system has been developed which allows developers to consider the performance impact of implementational considerations in high-performance software (such as whether to use arrays or linked lists) in a machine-independent manner prior to the writing of code [6].

2. Increasing integration of dynamics modelling tools with other CASE tools. An early effort [3] focussed on the production of SIMSCRIPT simulations from system descriptions in a variant of the problem statement language [20] (PSL). More recently, Cadre Technologies has introduced CASE tools such as Teamwork/SIM [12] with dynamic analysis capabilities via system simulation. These capabilities are distinct from diagram-completeness and static requirements analysis functions offered by numerous CASE vendors [10].

3. The use of graphically-oriented and integrated environments for IS dynamics modelling, and modelling and simulation in general. NEST [16] exemplifies the state-of-the-art in highly-functional, graphically-oriented communications network simulators. QASE RT supports the graphical depiction of software running on hardware. The simulation model development environment (SMDE) [2] exemplifies, general-purpose, graphical, and integrated simulation environments.

4. The codifying of simulation modelling knowledge into computer-based systems. In all of the above-mentioned tools, knowledge of the modelling task is embedded in the system and is the instrument by which these tools are more effective than using a simulation 4GL such as SIMSCRIPT or GPSS. Simulation modelling knowledge is encoded more explicitly in the on-line help system for SMDE [7] and in expert systems to aid in the statistical issues in simulation. One expert system [13] provides support including aid in determining simulation duration and number of replications, assessing the impact of initial conditions, constructing confidence intervals, comparing output from alternative designs, and estimating response/input variable relationships.

A prototype CASE/simulation system has been developed which automatically produces stochastic discrete-event-based simulation models from data flow diagrams (DFDs) and additional information regarding the performance of system components [21, 22, 23]. DFDs are selected because they are graphical, supported by current CASE technology, and familiar to IS analysts. Each DFD process is viewed as a server/queue in a queuing network. The prototype provides simulation in the context of an integrated simulation environment under the X Window System[1] graphical user interface. Intelligent help is provided in the formulation of simulation run parameters for the method of replications, and in interpretation of stochastic simulation output. Figure 1 illustrates major features of the prototype.

1.2. Research Questions

Several questions can be asked of a system intended to deliver design performance modelling technology to IS analysts:

Q1. Does the system model all necessary aspects of IS design dynamics to support the decision-making of analysts?

Q2. Does the system improve the convenience with which IS design dynamics can be evaluated?

Q3. Does the system improve the accuracy of IS design dynamics evaluation?

Q1 can best be answered by application of CASE/simulation system to a broad spectrum of IS problems. The prototype has been applied to a variety of fairly-standard IS designs [21, 22, 23], providing some demonstration of the feasibility of the approach. With respect to Q2, the prototype provides automated simulation from DFDs annotated with component-performance information. Since the drawing of DFDs is a normal part of IS development, a model of IS dynamics which utilizes the information in DFDs should be more convenient to the IS analyst than a model that requires re-coding of that information in a simulation 4GL or a specialized notation. Q3 seems especially amenable to being addressed directly via controlled experimentation.

Rather than simply answering Q3 for a given software system, it would be of greater theoretical interest to determine what models are most useful in improving the accuracy with which analysts evaluate the dynamics of IS designs. In practice, however, it is not realistic to deliver models to a decision-maker without some mode of presentation, such as a particular software environment. Additionally, in a software environment providing access to a simulation model (i.e., a *simulation environment*), it is difficult to draw a sharp distinction between aspects of the environment which are providing the simulation model and aspects of the environment which are providing help in the use of the model and interpretation of its output.

In light of the theoretical and practical considerations, the following two research questions are posed as questions about a software environment such as the prototype CASE/simulation system:

[1]X Window System and X11 are trademarks of the Massachusetts Institute of Technology.

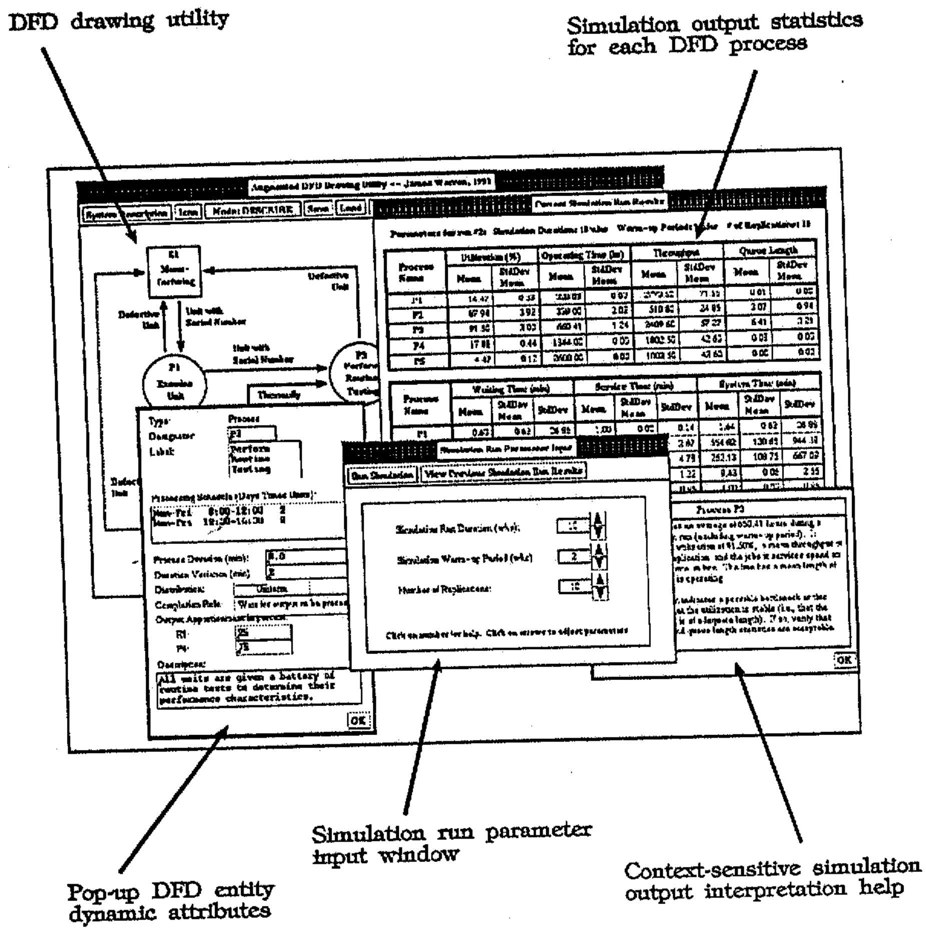

figure 1 The prototype IS design dynamics simulator runs under X11 on an HP9000 workstation.

Q4. Does a software environment providing queuing network simulation generated automatically from DFDs augmented with dynamics attributes, including expert help on simulation experiment design and output analysis, allow IS analysts to evaluate more accurately the dynamic performance of IS designs than an environment providing DFDs augmented with dynamics attributes but lacking simulation capability?

Q5. Which, if any, features of a software environment providing queuing network simulation generated automatically from DFDs augmented with dynamics attributes, including expert help on simulation experiment design and output analysis, are associated with more accurate evaluation of the dynamic performance of IS designs?

The authors' current research objective is to address questions Q4 and Q5. The results of the pilot study: (a) provide some support for a positive response to Q4 (i.e., that CASE/simulation does lead to more accurate assessment of IS design dynamics), and (b) point the way to further prototype developments and experiments, leading to deepening understanding of the methods of decision support in this domain. The method and results of the pilot study, and the implications of the pilot study results are discussed in this paper. This research direction should lead to usable guidelines for developers of real-world CASE/simulation systems.

2. Method

An important feature of the prototype CASE/simulation system is that its software environment can be invoked in one of two modes: (1) a mode which allows viewing of DFDs augmented with dynamics attributes, but provides no simulation capability; or, (2) a mode which, in addition to the DFD-viewing of mode 1, provides queuing network-based simulation of IS designs, including expert help on simulation experiment design and output analysis. From this feature of the prototype, we define two experimental treatment levels.

Level A. Subjects answer questions about IS designs using the prototype in mode 1, where they are provided with DFDs annotated with dynamics information, but have no simulation capability.

Level B. Subjects answer questions about IS designs using the prototype in mode 2, where they are provided with DFDs annotated with dynamics information, queueing network-based simulation of the IS design represented in the DFD, and expert help on simulation experiment design and output analysis.

Subjects from a population of IS graduate students answer a series of questions about the dynamics of IS designs. The subjects attempt to address various sets of questions at the two treatment levels. The observed mean difference in accuracy between treatment levels is expected to lead to the rejection of the null hypothesis that IS-dynamics evaluation accuracy measures are equal for treatment levels A and B. The usage of the software environment is also observed to support analysis of the association of aspects of tool usage (defined in section 2.1) with accuracy.

A repeated-measures design is used where each subject responds to each treatment level. Each subject analyzes and answers dynamics questions on four IS design cases, two cases at each level.

2.1 Dependent Measures

The dependent measure for accuracy in assessment of IS dynamics is determined via subject responses to questions regarding the dynamics of IS designs. Each response receives an evaluation accuracy measure between 0 (poorest accuracy) and 1 (perfect accuracy). Eight questions are asked regarding each of four cases, resulting in a score between 0 and 8 for each case. One of the four cases and its associated dynamics questions is given in appendix B.

As subjects work to answer dynamics question, *usage variables* are collected to characterize the manner in which the software system is utilized. These variables include: (a) the number of simulation runs conducted, (b) the total duration of the simulation runs (in simulated weeks), (c) the total warm-up period duration (in simulated weeks), (d) the total number of replications, (e) the number of times the simulation run parameter help is consulted, (f) the number of times the simulation output help is consulted, (g) and the number of times the DFD viewer annotation is consulted.

After each case, subjects rate their confidence in the accuracy of their responses to the dynamics questions on a seven-point scale.

2.1.1. Cases

The cases are evenly balanced between service and manufacturing. The first two dynamics cases (including the case given in appendix A) are based on illustrated cases in the IS textbook by Saldarini [15]. These cases concern a video-tape rental store and are representative of service-oriented systems. The second pair of cases concern a small high-tech manufacturing operation and are drawn from the author's own experience.

All of the cases are somewhat modified from their original forms. In particular, the cases are simplified to make the assessment of their dynamics more feasible within the limiting time constraints given to the experimental subjects. Also, many of the dynamic parameters of the cases (arrival rates and service rates) are fabricated to produce a system with stable performance characteristics and are not based in empirical investigation of the real systems.

2.1.2. Dynamics Questions

Eight dynamics questions are formulated for each case. The questions can be separated into four categories along two dimensions (see table 1). These dimensions are based upon the type of information required to answer the questions. A first dimension of dynamics question is *expected* versus *emergent*. *Expected* questions can be answered using straight-forward analysis of the expected values of parameters of the queuing systems. *Emergent* questions require the information gathered during a simulation experiment in order to formulate an answer. The second dimension is *simple* versus *composite*. A *simple* question requires the analysis of only a single server/queue system, whereas a *composite* question requires consideration of multiple server/queues. Each dynamics question can be categorized as *expected-simple*, *expected-composite*, *emergent-simple*, or *emergent-composite*. Two questions of each type are presented for each dynamics case.

In the illustrative case in appendix A, the return component of a video tape rental system, the first dynamics question concerns throughput. This is *expected-simple*, because the question can be answered through simple multiplication (arrival-rate times time-period) and requires only the consideration of one queuing system. Question 3, concerning utilization, is *expected-composite*, because, although it can be solved using a simple set of multiplications and additions, it requires consideration of multiple server/queues (P2.1 and P2.2). The fifth and seventh questions involve the system times of jobs. These are both *emergent*, because the system time is effectively intractable to calculate analytically, requiring use of simulation results[2]. Question 5 is *emergent-simple*, because only process P2.1 (a single server/queue system) must be considered once simulation results are utilized. The seventh question concerns

[2]Analytic calculation of the expected system time of a job in a queuing system is not intractable if proper simplifying assumptions are met (such as Poisson arrival and service processes), but these conditions are never met for the system time questions in the dynamics cases. While the questions would give way to a rigorous analysis, solution without resort to the simulation results requires a knowledge of analytical queuing theory well beyond the presentation in an introductory management science textbook, and outside the expectation for information systems professionals.

the processing sequence of customer files; information from multiple server/queue systems must be considered, and therefore it is *emergent-composite*.

	Single queuing system	Multiple queuing systems
Simple analytic reasoning	Expected-simple	Expected-composite
Simulation results	Emergent-simple	Emergent-composite

table 1 Four types of dynamics questions along two dimensions.

Dynamics questions are chosen to relate to concerns likely to arise due to the performance requirements of the analyzed system. For example, in the video tape return case, the questions focus on how long customers wait, how the clerk's time is spent, and how quickly inventory is returned to stock shelves. The responses to these questions characterize the system dynamics relevant to the adequacy of the design and the potential value of introducing further automation.

2.1.3. Data Transformation

The response to a dynamics question is transformed to obtain an accuracy measure between zero and one. The accuracy measure is:

$$score = \left(1 - \left(\frac{\varepsilon}{\varepsilon + min(\mathcal{R}, \hat{\mu})}\right)\right)^2 \qquad (2.1)$$

where ε is a measure of the error in the subject's response, $\mathcal{R}$ is the subject's response, and $\hat{\mu}$ is an estimate of the true value.

Questions of the expected type can be answered exactly using simple analytical techniques, whereas questions of the emergent type require use of stochastic simulation output in the formulation of a response. For emergent-type questions, any response within the 95% confidence interval offered by the simulation environment for 10 replications of a 10-week simulation run[3] should be considered perfectly accurate (assigned an accuracy measure of unity). For these questions:

$$\bar{\varepsilon} = \begin{cases} 0; & |\mathcal{R} - \hat{\mu}| < \delta_{95} \\ |\mathcal{R} - \hat{\mu}|; & else \end{cases} \qquad (2.2)$$

where δ_{95} is the half-width of the 95% confidence interval of the estimated actual value and $\hat{\mu}$ is the best estimate of the actual value (the center of the confidence interval).

[3]An arbitrary standard for an extensive simulation experiment.

For expected-type questions, $\mathcal{E}$ is simply $|\mathcal{R} - \hat{\mu}|$ where $\hat{\mu}$ is the exact analytically-determined correct response.

2.2. Subjects

Eleven IS graduate student volunteers from the University of Maryland are used in the present study. Only students that have passed the departmental comprehensive examination are accepted for the study. These students have a tested knowledge of IS technologies including a three-credit-hour graduate-level course in modelling and simulation. The volunteers have a mean of 12 years professional experience with information systems.

2.3. Apparatus

The primary experimental apparatus is the prototype CASE/simulation system software and associated hardware. The experiment is conducted in the University of Maryland Baltimore County Department of Information Systems Unix Lab using two HP 9000 series 375 workstations with 19-inch color screens running X11. Four monochrome X terminals are used in the introductory session, but in data-collecting sessions. The experimental environment is invoked automatically when subjects log on to the workstations.

At each experimental level, subjects are provided with two windows: (1) a DFD augmented with dynamics information describing the IS-dynamics case, and (2) a question-asking window querying the user about the dynamics of the IS design. Subjects are also provided with xcalc (a standard X11 utility which emulates a pocket-calculator), and paper and pencil. These two windows provide the subjects with the IS dynamics cases. In treatment level B subjects also receive the "Simulation Run Parameter Input" window (see figure 1), which provides access to IS design simulation and expert help in simulation usage and interpretation.

2.4. Procedure

Participation in the study takes the form of an a two-hour introductory session and two sixty-five-minute problem-solving sessions to be performed within one week of the introductory session. The introductory session has three components: (1) an administrative period, (2) a lecture to introduce key concepts, and (3) an introduction to the apparatus (hardware and software). During the administrative period, subjects read and sign an "Informed Consent" form and fill out a "Participant Name Form" and a "Professional Profile" form. The informed consent form explains the general outline of the experimental procedure to the subjects. The participant name form provides a confidential linkage between the subject number (that will be used in lieu of subject names in all analysis of the data) and the subject name for the purposes of experimental administration, and to allow subjects to review their performance at a later date, if they so desire. The professional profile form provides a summary of the professional experience level of subjects (not linked to subject name or number) for purposes of reporting results.

The introductory lecture is intended to convey four major topics:

1. The nature of simulation: ***simulation*** and ***modelling*** are defined, the concept of ***queuing systems*** is introduced, and the probabilistic nature of simulation results is emphasized.

2. Simulation run parameters for the method of replications (run durations, warm-up period, and number of replications) are explained.

3. Simulation output statistics (throughput, utilization, operating time, waiting time, service time, and system time) are defined and explained.

4. The model of DFD processes as queuing systems is discussed and the annotation of DFDs for simulation is demonstrated. The concept is illustrated via an investment system example which is carried into the introduction to the hardware/software system.

The introductory lecture closely follows the contents of a handout which subjects may use throughout the workshop. This lecture takes about forty-five minutes.

After a short break, the focus of the session turns to the apparatus. Subjects are shown how to log onto the UNIX workstations and how to invoke the CASE/simulation prototype with an introductory case based on the investment system example. The subjects are instructed in the use of the augmented DFD viewer and question-answering window. Subjects are then guided through the use of the simulator to examine the introductory case. At the end of the introductory session, subjects are encouraged to fill out an "Introductory Session Problems and Suggestions" form.

In each problem-solving session, subjects are presented with a description of an organization, and DFDs and corresponding data dictionary entries (augmented with dynamics information) for two component systems of that organization. Subjects are given 10 minutes to peruse these materials. After reading about the organization, subjects are presented with a set of eight IS design-dynamics questions related to one of the two component systems. Subjects invoke the X11 problem-solving environment for their case using a point-and-click interface. Two windows appear if the subject is receiving level A treatment (the "Augmented DFD Viewer" and the "Questions" windows); three windows for level B treatment (where the "Simulation Run Parameters Input" window is added to the windows in level A).

Subjects spend 20 minutes analyzing the case with the tools available to answer the dynamics questions on screen. The responses to the dynamics questions are recorded in a separate file for each subject at the end of the 20-minute session. Usage variables are also recorded at the end of the session.

At the end of the 20-minute analysis period, subjects are given 5 minutes to fill out a "Problems and Suggestions" form. On this form they report their level of confidence in their responses to the questions, and are encouraged to describe any difficulties they encountered in understanding the case, using the tool, or understanding the expert-help functions. After a five-minute break, subjects are presented with another set of eight questions (regarding the other component system of the organization), given 20 minutes to answer the dynamics questions, and 5 minutes to fill out another "Problems and Suggestions" form. At the end of each sixty-five-minute problem-solving session, subjects are given the worked solutions to the cases they had been analyzing.

2.4.1. Experimental Design

Learning effects over time are to be expected in this study, so the experimental design must address the order of presentation of the treatment levels. Table 2 shows the layout of a design which accounts for the carry-over effects of one treatment on the next [19]. Subjects are randomly assigned to one of two ordering groups. At time t_1, subjects randomly receive either case 1 or case 2 to analyze using either the treatment A environment (if they are in ordering group 0) or the treatment B environment (if they are in ordering group 1). At time t_2 (the second part of the first problem solving session), subjects receive either case 1 or 2, whichever they have not yet analyzed, using the treatment level A (if they are in ordering group 1) or B (if they are in ordering group 0) environment. In problem-solving session 2 (times t_3 and t_4) subjects analyze cases 3 and 4 in random order using treatment level A and B environments in the order determined by their ordering-group membership. In this design, the treatment effect is measured as the interaction of the ordering group membership and time.

	Time --->			
	Problem-solving Session 1		Problem-solving Session 2	
Ordering Group	t_1	t_2	t_3	t_4
Group 0	A	B	A	B
Group 1	B	A	B	A

table 2 Treatment levels by time for the study.

One particular mode of analysis is to consider two effects:

γ_1: the improvement in accuracy from the first case of a problem-solving session to the second, the intra-session learning effect.

γ_2: the improvement in accuracy from treatment level A to level B.

A single score is developed for each participant by adding the scores in those cases where the subject experienced treatment level B and subtracting from that sum the scores achieved in the cases performed at level A. That is, for a subject in ordering group 0, their score α_0 is:

$$\alpha_0 = s(t_2) + s(t_4) - s(t_1) - s(t_3) \qquad (2.3)$$

where $s(t_i)$ is the sum of the accuracy scores achieved at time i. The theoretical range for α_0 is from -16 to 16 if scores from all eight questions of each case are used, or from -8 to 8 if only the 4 emergent-type questions of each case are considered (i.e., the emergent-simple and emergent-composite questions). α_0 can be thought to measure the combined effect of learning and treatment, $\gamma_1 + \gamma_2$.

For a subject in ordering group 1, their score α_1 is:

$$\alpha_1 = s(t_1) + s(t_3) - s(t_2) - s(t_4) \quad (2.4)$$

α_1 can be thought to measure the treatment effect minus the learning effect, $\gamma_1 - \gamma_2$.

This leads to the following statistical null hypothesis:

$\mathbf{H_0}$: $\gamma_2 = (\alpha_0 + \alpha_1) / 2 = 0$.

This hypothesis is tested for: (a) the case where the $s(t_i)$'s are based on all eight dynamics questions, ranging from 0 to 8; and (b) the case where only the emergent-type dynamics question are considered in the $s(t_i)$'s, thereby giving a range from 0 to 4.

Note that, theoretically, the scores α_0 and α_1 are not normally distributed, because they have a limited possible range of values. As will be shown is the "Results" section, however, no observation approaches the limits of the range; the statistical null hypothesis of normality is not rejected.

3. RESULTS

In the pilot study, H_0 is not rejected using all dynamics questions. H_0 is rejected at the 2% level[4], however, for the emergent-type scores only. A significant correlation is found between the subjects' self-reported confidence rating and the treatment level.

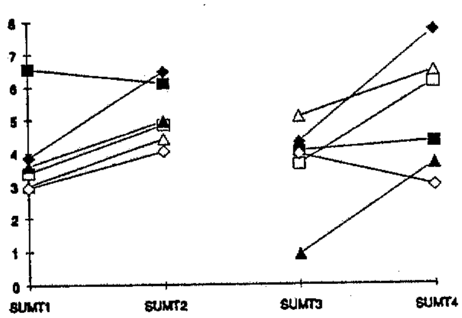

figure 2 Sum of the scores on all eight questions by time for subjects of ordering group 0.

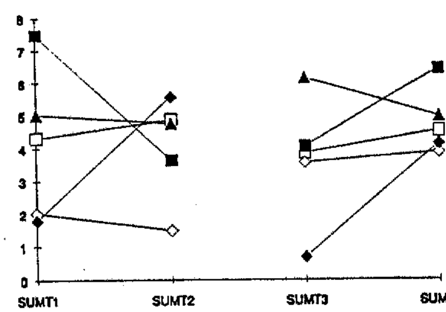

figure 3 Sum of the scores on all eight questions by time for subjects of ordering group 1.

3.1. Treatment Effect on Response Accuracy

Subject responses to dynamics questions are transformed into scores in the range [0-1] by the method given in section 2.1.4. Figures 2 and 3 graph the sum of the scores on all 8 dynamics

[4]Note that the pilot study was *exploratory* in nature, in that numerous hypotheses were considered and tested. Hence, the significance level of a test in the pilot study is merely a numerical indicator, and is not intended to be presented as proof of any theory.

questions for each of the four cases by time for ordering groups 0 and 1, respectively. The scores attained by an individual in the same session are joined by line segments to emphasize the slope. Figures 4 and 5 graph the sum of the scores on emergent-type questions only (questions 5 through 8) for each of the four cases for ordering groups 0 and 1, respectively. Ordering groups and levels are defined as in section 2.1.4.

Figure 6 graphs the interaction of ordering group and time for the means of the scores summing over all 8 questions. Figure 7 graphs the interaction of ordering group and time for the means of the scores summing over the four emergent-type questions only. One can see that the treatment effect (visible as the difference in slopes between ordering groups) is weaker with all eight scores than with emergent scores only.

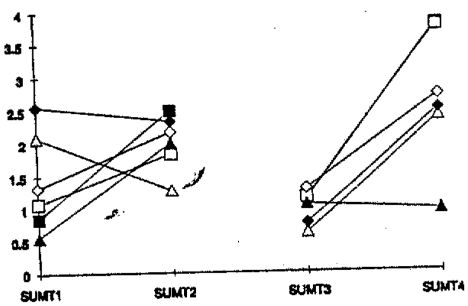

figure 4 Sum of the scores on the four emergent-type questions by time for subjects of ordering group 0.

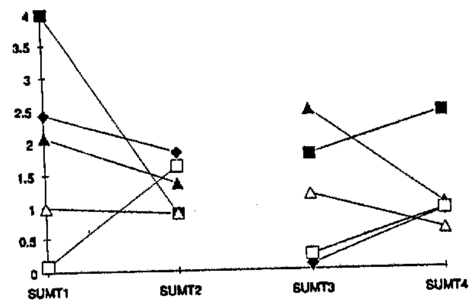

figure 5 Sum of the scores on the four emergent-type questions by time for subjects of ordering group 1

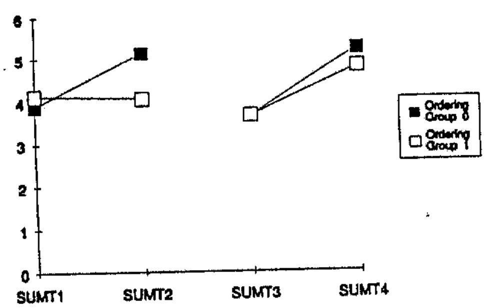

figure 6 Interaction of time and ordering group on the sum of the scores on all eight questions.

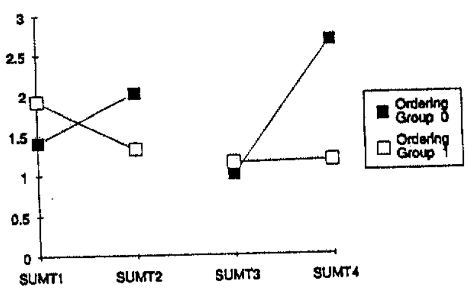

figure 7 Interaction of time and ordering group on the sum of the scores on the four emergent-type questions.

Summing the scores at time 1 and time 3 (the cases worked at the beginning of each problem-solving session) and the scores at time 2 and time 4, produces two measures for each subject. For ordering group 0, the first of these two measures is the sum of scores at treatment level A and the second is at treatment level B. For ordering group 1, the second measure is the sum of scores at level A, and the first measure is level B. The mode of analysis described in section 2.4.1 analyzes the differences of these two measures to assess the treatment effect. Figure 8 graphs the interaction of time (beginning-of-session versus end-of-session) by ordering group for these measures on the emergent-type questions.

The treatment effect γ_2 is equivalent to the ordering group by time interaction (within subjects) and is found to be significantly different from zero at the 1.63% level. Note that the treatment effect is not significantly different from zero when scores on all eight dynamics questions of each case are considered.

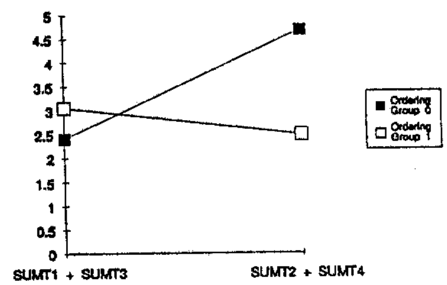

figure 8 Interaction of time and ordering group on the sum of the scores on four emergent-type questions, where t_1 and t_3, and t_2 and t_4 scores are combined.

3.2 Software System Usage Analysis

Figure 9 illustrates the superior accuracy in response to emergent-type questions demonstrated by subjects that conduct more simulation runs when simulation is available (i.e., in treatment level B). The difference is significant at the 5% level. Due to small sample size, no other significant models of accuracy as a function of usage pattern are formulated.

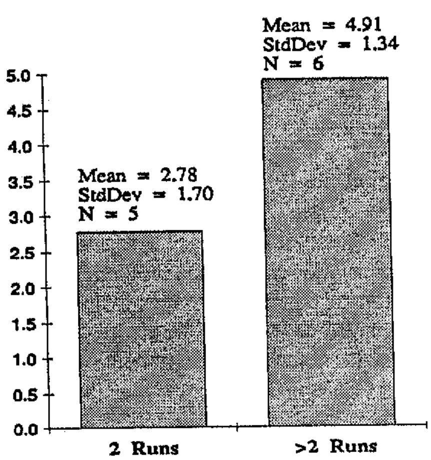

figure 9 Total accuracy scores on emergent-type questions for cases where subjects had access to simulation (treatment level B) by total number of simulation runs conducted.

3.3 Survey Results

Responses to the multiple choice question on the "Problems and Suggestions" form regarding confidence in accuracy of responses are coded as integers in the range 1 to 7. Table 3 shows the results of correlation analysis between the confidence rating and the accuracy score (summed over all eight questions), time when case is worked (1 for beginning of session one through 4 for end of session two), and treatment level (0 for treatment level A, 1 for B). These values are formulated by counting each case for each subject as a separate observation (i.e., 11 x 4 = 44 observations). The confidence rating is coded such that 1 is most confident and 7 is least confident. Using this method, accuracy score and confidence rating are significantly associated (3.66% level). Practice has a marginally-significant association (5.39% level) with confidence. Confidence rating is significantly associated (0.67% level) with treatment level.

Variable	Correlation Coefficient	Significance of Coefficient
Accuracy Score (summed over all eight questions)	-0.316*	3.66%
Time Case is Worked (1 through 4)	-0.293	5.39%
Treatment Level (A or B)	-0.403	0.67%

* Pearson correlation coefficient.

Table 3. Correlation analysis of confidence rating from "Problems and Suggestions" form. (All correlation coefficients are Spearman unless otherwise noted.)

4. CONCLUSIONS

The primary finding of the pilot study is that the CASE/simulation system appears to be a significant aid to accuracy in response to emergent-type IS design dynamics questions. Emergent-type questions concern situations where queuing is a factor. Queuing statistics are central to quantifying customer waiting times, manufacturing lead times, numbers of jobs in progress, the time for detection of errors and defects, the time for processing a request, and the time merchandise lays idle before being made available to customers -- quantities critical to the acceptability of modern, non-batch information systems. Preventing errors in the assessment of queuing-related aspects of IS designs constitutes a highly-desirable goal.

Due to the exploratory nature of the pilot study, the authors do not consider the findings statistically significant until they are replicated in future research. The results, however, are encouraging for the potential benefits of CASE/simulation systems.

Several additional features of the results are noteworthy:

1. The lack of a significant treatment effect on all eight questions combined. Further investigation of this lack of effectiveness seems in order. Logically, given that the treatment improves scores on emergent-type questions, then it must actually decrease scores on expected-type questions in order for their to be no positive impact on the combined expected-type and emergent-type scores. This seems unlikely and points out the combination of two weaknesses in the pilot study. First, the pilot study subjects have an exceptionally strong background (versus average IS professionals) in modelling concepts, and are able to answer expected-type questions quite accurately without the aid of computer simulation, thus decreasing the possible effect size. Second, small sample size and high intra-subject variance resulted in low statistical power (see section 4.1).

2. The dispersion of the scores for ordering group 1 at the beginning of the first problem-solving session. Figures 3 and 5 clearly reveal that subjects which began problem-solving sessions at treatment level B experienced exceptional variance in scores, particularly in the first problem-solving session. One may guess that some subjects experiencing the more complex software environment available .at level B experienced confusion and subsequently reduced scores. This is a type of effect for which the experimental design does not compensate, and is a source of reduced power and error. Future experiments should better prepare subjects before beginning the problem-solving sessions. If a reasonable level of preparation does not reduce this effect then we may be forced to the conclusion that the environment is confusing to IS professionals.

3. The superior accuracy in response to emergent-type questions by those subjects running more than one simulation, on the average, when simulation is available. This difference appeals to the notion that subjects that take a more iterative approach to simulation usage form more accurate responses to dynamics questions. This result, if confirmed in future experiments, supports the notion that the architecture of a CASE/simulation system should encourage iterative problem-solving.

4. The significant association of confidence with the availability of simulation capability. This correlation is not surprising, and is among the common decision support system effects noted in the literature [17]. Ironically, the treatment boosted confidence far less ambiguously than performance. It is the authors' view that this variety of result reaffirms the need for careful experimental evaluation of new tools and techniques. Engineering systems to meet the expressed desires of users is relatively easy in comparison to assessing what innovations actually improve users' performance.

4.1 Further Research

The pilot study is intended to lay the foundation for further research in CASE/simulation systems. The pilot study findings indicate that the general experimental design is workable, and that a further experiment modeled on the pilot, with some refinements, should yield considerable useful information. High variance in scores and low mean scores for ordering group 1 subjects when using the simulation environment (treatment level B) precipitated the inability to reject the null hypothesis of no treatment effect on the accuracy measure summed over all eight questions. Also, lack of variance in simulation usage variables led to the formulation of a limited model of software system usage.

A future study, described in detail in [24], attempts to overcome the weaknesses of the pilot by:

1. Better preparing subjects during the introductory session with a more extensive example case, more time, and a greater emphasis on problem-solving techniques.

2. Conducting the study in the form of a one-day workshop, with problem-solving session in the same day as the introductory session, to intensify the learning experience for subjects.

3. Providing more time with each case, up from 20 minutes to 30 minutes to provide more opportunity for variance in software system usage.

4. Measuring additional aspects of tool usage including the times at which dynamics questions are answered.

Additionally, twice as many subjects will be used as in the pilot, increasing statistical power. Furthermore, the population for the pilot study, IS graduate students, threatens the generalization of results to the population of IS analysts, which, on the average, have had less exposure to simulation methods. The subjects in the future study will be drawn from IS professionals throughout the Baltimore-Washington area, a population which is less overtly biased toward special knowledge of simulation modelling.

The pilot study results can be used to estimate the sample size needed in a future study to achieve a 0.05 level of significance and a power of 0.80. For emergent scores only, the observed values of α_0 and α_1 are 2.285 and 0.574 with standard deviations of 1.306 and 1.910, respectively. This yields an estimate for γ_2, the treatment effect on accuracy, of 1.423 with a standard deviation of 1.157, and hence a treatment effect size, d, of 1.23. Using power tables for one-way ANOVA [9], with d = 1.23 and two levels, a power of 0.80 with a significance level of 0.05 can be achieved with 13 subjects per group, or a total of 26 subjects.

Helping subjects to achieve more consistent scores than those observed in the pilot study should decrease the needed sample size by increasing d. A sample size of between 20 and 25 subjects is expected to deliver a power of 0.80 at the 0.05 significance level in the future study.

ACKNOWLEDGEMENTS

Special thanks are owed to the graduate students who volunteered their valuable time to participate in the pilot study.

REFERENCES

[1] Ammar, R., "A Computer Aided Design System to Develop High Performance Software", *Journal of Systems & Software*, Vol. 15, pp. 139-147, 1991.

[2] Balci, O., and R. Nance, "Simulation Model Development Environments: A Research Prototype", *Journal of the Operational Research Society*, Vol 38, No. 8, pp. 753-763, 1987.

[3] Boydstun, L., D. Teichroew, S. Spewak, Y. Yamamoto, and G. Starner, "Computer Aided Modelling of Information Systems", in: *Proceedings, IEEE COMPSAC80*, 1980.

[4] Chan, A., S. Danberg, S. Fox, W. Lin, A. Nori, and D. Ries, "Storage and Access Structures to Support a Semantic Data Model", in: *Proceedings of the 8th International Conference on Very Large Data Bases*, Very Large Database Endowment, 1982.

[5] Chan, A., U. Dayal, S. Fox, and D. Ries, "Supporting a Semantic Data Model in a Distributed Database System", in: *Proceedings of the 9th International Conference on Very Large Data Bases*, Very Large Database Endowment, 1983.

[6] Date, C., "On the Performance of Relational Database Systems", in: *Relational DataBase: Selected Readings*. Addison-Wesley, 1986.

[7] Frankel, V., and O. Balci, "An On-line Assistance System for the Simulation Model Development Environment", *International Journal of Man-Machine Studies*, Vol. 31, pp. 699-716, 1989.

[8]. Gore, A., "QASE to Configure Huge Systems", *MACWEEK*, Nov. 13, pp. 20, 1990.

[9] Maxwell, S., and H. Delaney, *Designing Experiments and Analyzing Data: A Model Comparison Perspective*, Wadsworth Publishing, 1990.

[10] McClure, C., "The CASE Experience", *Byte*, April, pp. 235-241, 1989.

[11] Mylopoulos, J., P. Bernstein, and H. Wong, "A Language Facility for Designing Database-Intensive Applications", *ACM Transaction on Database Systems*, Vol 5, No. 2, pp. 185-207, 1980.

[12] Pallatto, J.. "Cadre to Expand Suite of CASE Workstation Tools". *PC Week*, Nov. 26, 1990.

[13] Park, Y., and J. Mellichamp, "A Statistical Expert System for Simulation Analysis". *Proceedings, Summer Computer Simulation Conference*, 1990.

[14] Peckman, J., and F. Maryanski, "Semantic Data Models", *ACM Computing Surveys*, Vol. 20, No. 3, pp. 153-189, 1988.

[15] Saldarini, R., *Analysis and Design of Business Information Systems*, Macmillan, 1989.

[16] Schwartz, A., Y. Yemini, and D. Bacon, "NEST: A Network Prototyping Testbed", *Communications of the ACM*, Vol. 33, No. 10, pp. 63-74, 1990.

[17] Sharda, R., S. Barr, and J. McDonnell, "Decision Support System Effectiveness: A Review and an Empirical Test", *Management Science*, Vol. 34, No. 2, pp. 139-159, 1988.

[18] Shipman, D., "The Functional Data Model and the Data Language DAPLEX", *ACM Transactions on Database Systems*, Vol 6, No. 1, pp. 140-173, 1981.

[19] Stevens, J., *Applied Multivariate Statistics for the Social Sciences*. Lawrence Erlbaum Associates, 1986.

[20] Teichroew, D., and E. Hershey, "PSL/PSA: A Computer-Aided Technique for Structured Documentation and Analysis of Information Processing Systems", *IEEE Transactions on Software Engineering*, Vol. 3, No. 1, pp. 41-48, 1977.

[21] Warren, J., J. Stott, and A. Norcio, "Stochastic Simulation of Information Systems Designs from Data Flow Diagrams", *Journal of Systems & Software*, to appear.

[22] Warren, J., J. Stott, and A. Norcio, "A Prototype for Including Simulation of IS Dynamics in CASE Environments", in: *Proceedings, Twenty-Fifth Hawaii International Conference on System Sciences*, 1992.

[23] Warren, J., and J. Stott, "CASE/Simulation: Making Performance Evaluation a Normal Part of IS Development" *Proceedings, Second International Working Conference on Dynamic Modelling of Information Systems*, 1991.

[24] Warren, J. *CASE/Simulation Systems*, Doctoral dissertation at the University of Maryland, Department of Information Systems, Baltimore, 1992.

APPENDIX A. A SAMPLE CASE

For each case, subjects are presented with a system description, DFD, and a data dictionary. After perusing these material, they are presented with eight dynamics questions to answer to the best of their ability.

A.1. System Description

The video tape rental store currently uses a primarily manual information system in its dealings with customers. For each member, a membership envelope is maintained in a customer file. Whenever a member rents or returns video tapes, their membership envelope is pulled from the customer file and updated with information regarding the transaction. When customers rent tapes, the rental clerk fills out a rental agreement form. Copies of the form are given to the customer, placed in the order book, and placed in the customer's membership envelope. When the customer returns the rented inventory, the customer presents their copy of the rental agreement and pays for the rental. The return clerk removes the copy of the rental agreement from the membership envelope and disposes of it, initials the copy in the order book, and initials and returns the customer's copy of the agreement to act as a receipt.

You should have a DFD and data dictionary entries for the return component of the video tape rental system. All entities in the return component are numbered 2.*x*. Customers arrive to the return desk randomly (inter-arrival times following the exponential distribution) at an average rate of 1 every 4 minutes during the operating hours, noon to 8 pm weekdays. Customers present the return clerk with their pink customer copy of the rental agreement and payment along with the rental inventory. The rental clerk accepts these items and pulls the customer's membership envelope from the customer file (requiring a total of one minute, distributed exponentially). The clerk then performs the return processing, removing the gold copy of the rental agreement from the membership envelope and disposing of it, placing the payment in the cash drawer, returning the inventory to the shelving bin, pulling the appropriate white copy of the rental agreement from the order book, initialing it, and returning it to the order book, and initialing and returning the customer's pink copy. This return processing requires a mean of 1.75 minutes, distributed normally with a standard deviation of 0.5 minutes. At the end of the day, the return clerk re-shelves the inventory in the shelving bin, returning it to the stock shelves (requiring 0.3 to 0.7 minutes per customer return). After re-shelving inventory, the clerk re-files the membership envelopes in the customer file (requiring 30 seconds per job, distributed exponentially).

A.2. Data Dictionary

Data dictionary entries annotated with dynamics information, such as this entry for process P2.1, are given for each DFD process, external entity, data store, and data flow corresponding to the elements shown in the DFD in figure A.1.

```
NAME:          P2.1: Look-up Membership
INPUT:         E2.1: Customer
               D2.1: Customer File
OUTPUT:        P2.2: Process Return
               D2.2: Membership Envelope
JOB DURATION:1.00 minutes/job, distributed exponentially.  Processor waits
               for completion of job at output process before continuing to
               next job.
PROCESSORS:    1 unit:   Mon-Fri  12:00-20:00 and idle
DESCRIPTION:   Return clerk accepts inventory, form, and payment from
               customer and pulls their membership envelope from the customer
               file.
```

A.3. Dynamics Questions

1. How many membership envelopes does the clerk re-file at the end of each day?
2. How long does the clerk work each week at placing returned inventory on the stock shelves?
3. What is the utilization (in percent) of the return clerk at P2.1?
4. How long does a customer spend being serviced at the return desk (excluding time waiting in line)?
5. How long does a customer spend at the return desk (including time waiting in line)?
6. How long is the line at the return desk (excluding the customer being serviced)?
7. How much time elapses between the time a customer's membership envelope is pulled from the customer file (at the beginning of customer service at P2.1) and the re-filing of the envelope in the customer file?
8. How much time elapses between the time a customer begins service at the return desk (P2.1) and the placement of the returned inventory on the stock shelves?

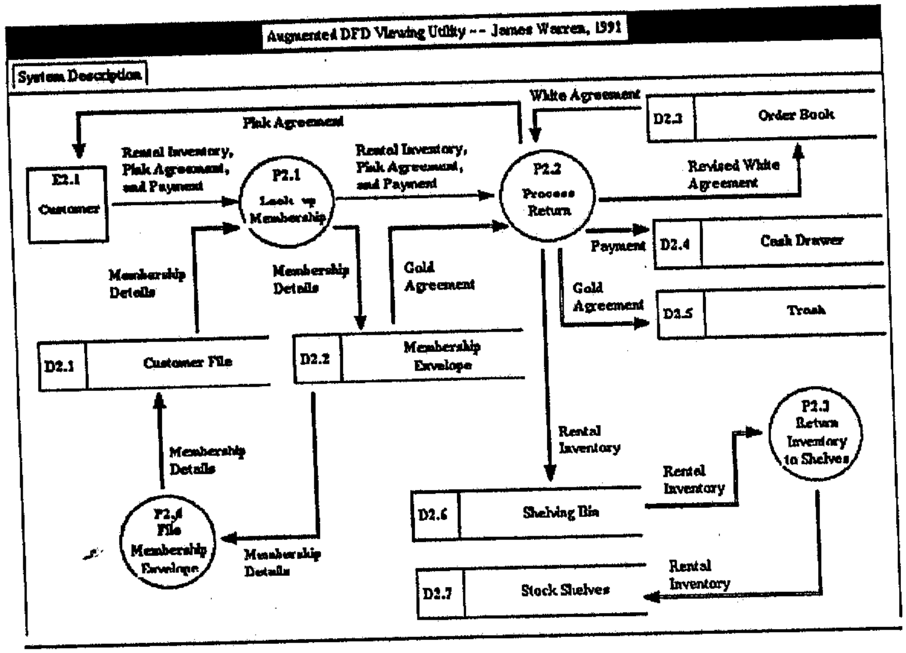

figure A.1 DFD of the return component of a video tape rental system.