Success Surrogates in Representational Decision Support Systems

Roger McHaney
Kansas State University, USA

INTRODUCTION

Rapid and frequent organizational change has been a hallmark of business environments in the past two decades. Frequently, technology and new software development are embraced as aspects of complex strategies and tactical plans. Without sufficient analysis, the unforeseen consequences of change can result in unexpected disruptions and the loss of productivity. In order to better control these contingencies, modern managers often employ a variety of decision support aids. One such aid, classified as a representational decision support system, is discrete event simulation (DES).

BACKGROUND

In its purest form, DES is considered to be a branch of applied mathematics. Its considerable popularity is due in part to the availability of computers and improvements in simulation languages and simulator software packages. DES is often the technique of choice when standard analytical or mathematical methodologies become too difficult to develop. Using a computer to imitate the operations of a real-world process requires a set of assumptions taking the form of logical relationships that are shaped into a model. These models assess the impact of randomly occurring events. Experimental designs are developed and the model manipulated to enable the analyst to understand the dynamics of the system. The model is evaluated numerically over a period of time and output data is gathered to estimate the true characteristics of the actual system. The collected data is interpreted with statistics allowing formulation of inferences as to the true characteristics of the system. Table 1 lists primary features of a DES application.

While the value of DES in organizational settings has been accepted and is evidenced by the varied and growing market of related products, not every DES application is suited for every problem domain. For this reason, information systems researchers have worked to identify salient characteristics of DES and its usage and then measure the relationship between its application and successful organizational outcomes. These studies have been conducted in different ways with focuses on independent and dependent variables.

INDEPENDENT VARIABLE RESEARCH

Much DES research has focused on identifying and evaluating software and project characteristics (independent variables) and then producing recommendations that

Table 1. DES features

Feature	Description
Statistics Collection	Tools which gather data for purposes of inferential statistics about the model
Resource Modeling	A means for the representation of a constrained resource in the model
Transaction	A means for representation of the participants in the simulatior model
Simulation Clock	Tools for analysis and step processing of the coded model
Random Number Generators	A means for producing random number streams for randomization of events within the simulation model
Model Frameworks	Generalized frameworks for the rapid development of a model

Table 2. List of DES success factors

Factor 1: Software Characteristics
Factor 2: Operational Cost Characteristics
Factor 3: Software Environment Characteristics
Factor 4: Simulation Software Output Characteristics
Factor 5: Organizational Support Characteristics
Factor 6: Initial Investment Cost Characteristics
Factor 7: Task Characteristics

either suggest how a project can be successfully implemented or how failure can be avoided. A variety of useful practitioner-focused articles have been published in this area (Banks, 1991; Law & Haider, 1989; Swain, 2003). These studies provide recommendations, based in part on first-hand experience, of consultants and simulation analysts. In many cases, these recommendations list the packages currently available and focus on the pros and cons of each.

Academic studies of independent variables have also been conducted. In one of the first studies using information systems as basis for DES, McHaney and Cronan (2000) extended a framework of general decision support system (DSS) success factors identified by Guimaraes, Igbaria, and Lu (1992). The developed contingency model was theoretically derived from the simulation literature and empirically tested. The results indicated a seven-factor model that was structured as shown in Table 2.

Academic research by Robinson (1999) approached the problem from the opposite perspective and identified sources of simulation inaccuracy which may result in project failure. Other studies, such as one by McHaney and White (1998), developed a DES software selection framework which matched salient DES characteristics with software package features. The importance of evaluation and selection of appropriate DES packages was determined in relation to the success of simulation implementation. As might be expected, the choice of the wrong simulation package often correlated with simulation system failure. This study provided a set of criteria to be systematically considered when evaluating DES software. The taxonomy for simulation evaluation, together with importance ratings provided by the collective expertise of a large number of DES users, was used as a guideline in deciding the relative weighting to give various software package capabilities.

Other research (McHaney, White, & Heilman, 2002) attempted to develop an understanding of DES project success by determining which characteristics of a simulation project were more likely to be present in a successful simulation effort. Potential success factors were derived from the simulation literature and used to develop a questionnaire. Based on the findings, projects perceived as failing were often characterized by high costs, model size constraints, and slow software. Successful projects were characterized by teamwork, cooperation, mentoring, effective communication of outputs, high quality vendor documentation, easily understood software syntax, higher levels of analyst experience, and structured approaches to model development. By understanding the simulation process and characteristics of successful simulations, practitioners will find it easier to avoid common mistakes that can ruin a modeling effort.

DEPENDENT VARIABLE RESEARCH

The studies mentioned in the previous section focused on understanding independent variables defining successful representational decision support system applications. A problem with research that considers only the independent side of the equation is the lack of objective measures of whether suggested recommendations correlate with desired outcomes. This dilemma has been problematic throughout information systems research in general and has been the subject of academic debate (Delone & McLean, 1992). DES researchers recognize that the identification of a meaningful, reliable, and robust dependent variable is central to being able to conduct accurate comparisons between competing tools, techniques, software implementations, project approaches, and modeling perspectives.

A wide variety of dependent variables have been investigated in the broader fields of information systems (Delone & McLean, 1992) and decision support systems. Among these, DSS researchers have focused on success surrogates and investigated the possibility of assessing DSS success or failure. Representational DSS researchers have extended the use of these surrogates.

The first information system success surrogate validated in the context of DES was the end-user computing satisfaction instrument (Doll & Torkzadeh, 1988). The EUCS instrument, shown in Figure 1, is of particular interest because most applications of discrete event computer simulation can be categorized as end-user computing. McHaney and Cronan (1998) collected data from 411 participants using a variety of discrete event computer simulation software packages. The analysis indicated the instrument retained its psychometric properties and provided a valid success surrogate for end users beyond the introductory stages of using representational DSSs. The study established the use of information systems surrogate success measures for applied instruments. The results suggest EUCS can reliably and confidently be used in the investigation of competing tools, features, and

Related Content