Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 1995 Proceedings

Americas Conference on Information Systems (AMCIS)

8-25-1995

Information Requirements for Decision Support Systems: A Task Behavior Orientation

Glenn J. Browne
University of Maryland, Baltimore

Information Requirements for Decision Support Systems: A Task Behavior Orientation

Glenn J. Browne
Department of Information Systems
University of Maryland, Baltimore
Baltimore, MD 21228

Introduction

It has been argued often and convincingly that requirements determination is the most critical phase of information systems development (Byrd, Cossick, and Zmud, 1992; Dalal and Yadav, 1992). System misuse or disuse can often be traced to an inadequate requirements determination process. The majority of information systems developed for organizations require post-implementation revisions to meet users' needs (Wetherbe, 1991). Although information systems are expensive to develop initially, changes made after a system has been completed are much more expensive than making the same changes during the design process (Boehm, 1981). Consequently, research that can enlighten and improve the requirements elicitation process can make an important contribution to the quality and cost-effectiveness of software development efforts.

This paper proposes a task behavior-oriented approach to the determination of information requirements for the design of decision support systems (DSS). It is argued that the behavioral nature of DSS requires a shift from the data-driven requirements determination approaches used to construct transaction processing and similar systems. The current trend toward task performance-centered DSS in industry (Gery, 1995) also supports a shift in requirements determination focus.

Typical information requirements determination (IRD) methods include structured interviews, questionnaires, observation, and joint application design, among others (Whitten, Bentley, and Barlow, 1994). In most of these techniques, users are asked questions relating to goals, data, problems, and critical success factors, for example, and the answers are used to infer system requirements. However, these methods may not adequately specify the actual task performance behaviors in which users engage, as the methods are generally data-focused. Further, the level of requirements elicited through such methods is often too general to be of significant use to analysts attempting to understand behavior. Tools designed to elicit more specific task behaviors are likely to be of greater benefit, since such behaviors are better descriptors of user needs and arguably can lead to more complete system requirements (Keen, 1980). Because DSS are developed to support organizational tasks that decision makers perform, it is arguable that the requirements determination for such systems should more directly address users' performance of those tasks.

A Task Behavior-Focused Framework for Information Requirements Determination

To shift the focus of requirements determination to task behaviors, it is useful to frame IRD in terms of inputs, techniques, methods, tools, and outputs. The framework appears in Table 1. The particular technique(s), method(s), and tool(s) used depend on the task(s) to be supported by the DSS. Theoretically, any method can be used with any technique, and any tool with any method, although certain combinations may be more suitable than others from a practical standpoint. Three techniques are identified for eliciting task behavior requirements: job task analysis, workflow analysis, and decision process analysis. Job task analysis (Gagne, Briggs, and Wager, 1988) is a technique for analyzing the requirements of the task: What activities are or should be performed to successfully complete the task? Workflow analysis is a technique to model the user's flow of behavior in performing the task. Decision process analysis identifies the decision points in the user's workflow process, the places at which reasoning, judgments, and/or choices are necessary and at which improvements in the process can therefore be made.

Methods are used to implement the techniques. Methods include structured interviews, silent and interactive observation, document analysis, and protocol analysis, among others. A particularly useful method for analyzing task behaviors is interactive observation, which consists of the analyst interacting with the user while the latter performs his or her job. This method is discussed in more detail below.

Tools are specific ways of gathering requirements. Traditional tools for IRD include standardized questions (e.g., Critical Success Factors, Business Systems Planning, Ends-Means Analysis), various types of flowcharts (e.g., entity-relationship diagrams, data flow diagrams), and observation. Both standardized questions and flowcharts have weaknesses that are especially apparent in behavior-oriented systems. Typical standardized questions do not elicit enough detail to fully inform system design. Entity relationship diagrams are designed more for data modeling than for understanding behavior, and data flow diagrams capture a feature of the environment, flows of data, that is often of secondary importance in DSS. Observation suffers from a lack of good methods for organizing the observed behaviors. A number of tools have recently been created or adapted to improve the requirements elicitation dialog between analysts and users in DSS development. Such tools include, e.g., influence diagrams, conceptual model diagrams, and scenario building tools.

Outputs of behavior-focused IRD include such items as job process flow charts, decision rules and processes, users' mental models and metaphors, and position matrices (McGraw, 1994).

Methods and Tools for Improved IRD

For purposes of this abstract, one IRD method and one tool from the framework will be discussed. The method is interactive observation. As noted, behaviors can be described either through observing users or by questioning them as to their actions. Although interviews have been the primary method for IRD, interactive observations of individuals or groups, in which the analyst observes the user performing the task to be supported and prompts and probes at appropriate moments, can be very informative for determining

requirements. The main strength of the interview method is obviously the ability of the analyst to probe more deeply into areas of particular concern. However, observation is arguably a more comprehensive method of IRD than questioning approaches, since it can capture behaviors not anticipated by the systems analyst. Interactive observation combines the best aspects of the two methods to improve the outputs of IRD.

The tool introduced here is a behavior classification tool. The tool can be used with a variety of IRD methods. However, it seems particularly appropriate for observation, since, as noted, current observation methods suffer from a lack of tools for organizing the observed behaviors. Behavior classifications are commonly used in instructional systems development (see, e.g., Gagne, Briggs, and Wager, 1988) but are not as well known in information systems design. The classification tool proposed here relies on a theoretical conceptualization introduced by Hackman (1969) and further examined by Fleishman and Quaintance (1984). The conceptualization is based on four theoretical approaches to the study of task performance: a behavior description approach, a behavior requirements approach, an ability requirements approach, and a task characteristics approach. Only the behavior description approach is discussed in this abstract.

The behavior description approach requires that the behaviors people actually use in performing the designated task be described to allow relevant support mechanisms to be designed into the DSS. To operationalize this approach, the present research adapts a behavior classification scheme designed by Berliner, Angell, and Shearer (1964; See also Tuckman, 1992). The scheme is based on a series of descriptive active verbs, which attempt to capture the behaviors in which decision makers engage. The tool is intended to support various system functions. Therefore, descriptions are available to help determine interface, data, decision, and communication needs.

Support for the use of active verbs in DSS design was provided by Keen (1980). Keen noted that active verbs identify discrete intellectual operations of users in a task; such operations must be supported for the DSS to be useful. If a system function does not relate directly to some cognitive operation in the user's mind, the function will not be used. In the current context, the verbs will serve as a bridge between the cognitive operations and the required system functions.

Examples of proposed behavior descriptions appear in Table 2. They are categorized according to the types of information processes and system elements they seem most likely to support. The information processing classifications and specific behaviors listed by Berliner, Angell, and Shearer (1964) have been modified and supplemented here. The system needs have been added in the present context. Table 3 adds examples of specific system elements implied by various cataloged behaviors. For example, people engaged in a strategic planning task may be observed generating numerous alternatives; such an observation might lead to the inclusion of a brainstorming tool in a DSS designed to support this task. Similarly, observations of users attempting to estimate future sales based on a variety of criteria may suggest the need for a regression tool.

The behavior classification tool offers a rich context within which to catalog behavior. It helps the systems analyst to know what behaviors to look for and to organize what he sees. It then provides implications for system tools based on the observed behaviors. Thus, it provides a way to observe behavior intelligently and to link the behavior to tools. Hence, the value of this approach goes beyond theory by connecting behavior to design.

This paper is available in a longer version. References are available upon request.

Table 1
Task Behavior-Focused IRD Framework

Inputs	Techniques	Methods	Tools	Outputs
Task(s)	Job Task Analysis Workflow Analysis Decision Process Analysis	Structured Interviews Silent Observation Interactive Observation Focus Groups Surveys Document Analysis Protocol Analysis	Scenario Building Note Boards Influence Diagrams Conceptual Model Diagrams Event Analysis Flow Charts Standardized Questions Scenario Response Tasks Behavior Classifications Task Analysis Cognitive Maps Ad Hoc Questioning	e.g., Job Process Flow Charts

Table 2
Behavior Description Classification
(Adapted from Berliner, Angell, and Shearer, 1964)

User Processes	System Needs	Specific Behaviors
Perception	Interface	Detects, Discriminates, Identifies, Inspects, Locates, Reads, Receives, Scans, Surveys
Memory	Data	Recalls, Recognizes, Retains, Stores
Decision	Decision Support	Analyzes, Argues, Calculates, Categorizes, Chooses, Codes, Compares, Estimates, Generates, Imagines, Interpolates, Organizes, Plans, Predicts, Ranks, Rates, Tabulates, Votes
Communication	Communication	Advises, Answers, Directs, Indicates, Informs,

Instructs, Negotiates, Requests, Transmits

Table 3
Example Decision Behaviors and Support Tools

Decision-Making Behaviors	Possible Support Tools
Analyzes	ANOVA, Regression, Content Analysis
Compares	Rating, Ranking, Weighting
Estimates	Regression, Time Series
Generates	Analogy, Brainstorming, Scenario Building
Tabulates	Counter, Calculator, Spreadsheet