A GOAL-DRIVEN COGNITIVE MODEL OF THE SYSTEMS EVALUATION PROCESS

Kieran Mathieson
*Oakland University*

Terence Ryan
*Southern Illinois University, Edwardsville*

Follow this and additional works at: [http://aisel.aisnet.org/icis1991](http://aisel.aisnet.org/icis1991)
A GOAL-DRIVEN COGNITIVE MODEL OF THE SYSTEMS
EVALUATION PROCESS

Kieran Mathieson
Department of Decision and Information Systems
Oakland University

Terence Ryan
Department of Management Information Systems
Southern Illinois University, Edwardsville

ABSTRACT

Researchers have tried to build valid and reliable instruments for measuring users’ assessments of IS. However, focusing on instruments is a somewhat ineffective way of dealing with problems associated with the evaluation process itself. This paper presents a model of the process by which individuals evaluate an IS. The model emphasizes the importance of raters’ goals. To form an evaluation, raters seek information about an IS and compare it to their normative beliefs. The information they seek is influenced by their task support goals. Evaluations can be adjusted to serve raters’ social adjustment, value expressive, and protective goals. Empirical evidence supporting the model is discussed, along with the model’s implications for system developers. Some suggestions for future research are made.

1. INTRODUCTION

Users’ evaluations of information systems (IS) are important for two main reasons. First, user evaluations provide information to help guide development. In prototyping (Naumann and Jenkins 1982), user evaluations are a central part of the project control mechanism. A developer gives a prototype to a user for evaluation and changes the prototype based on the user’s suggestions for improvements. If the user’s ideas about the system are incorrect, then development could proceed in the wrong direction.

More recent development methodologies also emphasize the importance of user evaluations. For example, Martin’s (1991) Rapid Application Development methodology involves four different user teams who evaluate everything from requirements specifications to general design to the final system. Martin claims that users should actively participate in every phase of IS development.

The second reason that user evaluations are important is that the evaluations guide the behavior of the users themselves. For example, Davis, Bagozzi and Warshaw (1989) found that individuals’ intentions to use an IS are influenced by their evaluations of a system’s usefulness and ease of use. If their evaluations are not accurate, users could act incorrectly. On the one hand, they might waste resources on a system that initially seemed valuable but turned out not to be effective. On the other hand, users could miss an opportunity to improve performance by choosing not to use a system that has value.

Researchers have designed instruments to quantify users’ evaluations. Much of this work has focused on user information satisfaction (UIS) (e.g., Bailey and Pearson 1983; Doll and Torkzadeh 1988; Ives, Olson and Baroudi 1983; Jenkins and Ricketts 1985). Some researchers have identified limitations of UIS instruments. For instance, Galletta and Lederer (1989) note problems with scale units and origins, item heterogeneity, and reliability. Ryan and Bock (1990) have suggested that the ambiguity of items in UIS instruments complicates the interpretation of rater’s responses. However, some of these problems are not characteristics of the instruments per se, but of the way they are used. For instance, some users have little contact with IS developers and cannot evaluate them accurately, no matter what the instrument.

The current literature on users’ evaluation resembles the literature on employee performance appraisal as it was a decade ago. Traditionally, researchers focused their efforts on appraisal instruments, trying to reduce bias and increase reliability (Landy and Farr 1980). However, Ilgen and Favero (1985) noted that focusing on scale construction is a weak and indirect way of dealing with what are really problems in human judgment. Landy and Farr (1980) recommended a shift away from research on instruments to research aimed at understanding how raters evaluate employees. In response, researchers have developed models of the evaluation process (e.g., DeNisi, Caferty and Meglino 1984). Later work has reflected a concern for process issues. For instance, Pulakos (1986) developed successful rater training programs based on the implications of process models.

Given the importance of users’ evaluations of IS, similar efforts by MIS researchers are warranted. This paper introduces a model of the process by which an individual
evaluates an IS. The model's objectives are to (1) identify potential biases in the evaluation process and (2) provide a framework for research aimed at improving the utility of user evaluations to systems developers.

The next section presents the model. Empirical evidence supporting the model is then reviewed, along with recommendations for further tests. The model's implications for practicing system developers are identified. Finally, some suggestions for future research are made.

2. THE MODEL

This section first examines the context in which evaluations are formed. An overview of the model is then presented, followed by a discussion of the goals that a rater might pursue during an evaluation. Each of the model's components are then described.

2.1 The Evaluation Context

The model applies when a "rater" examines an "information system" and produces an "evaluation" that is reported to the "sponsor." A "rater" is a single individual, with some level of skill and experience with IS and with job-related tasks. An "information system" (IS) is a set of computer programs designed to help its user perform a set of tasks. During IS development, future users may be shown models that represent essential aspects of the final product. The models range from descriptions on paper to functioning prototypes. For the purposes of this discussion, the term "IS" includes these models.

An "evaluation" is an expression of a subset of a rater's beliefs about an IS. A "belief" is a probabilistic association between the IS (or some component of it) and an attribute or desired attribute (Fishbein and Ajzen 1975). Beliefs can be specific suggestions for changes, such as "the regional sales total should be highlighted," or general summaries, such as "the system is somewhat useful." General beliefs are inferred from specific beliefs. For example, if a rater believes that an IS supports only two out of three important tasks, the rater might infer that the IS is "somewhat useful."

Beliefs are formed from (1) information from direct experience, (2) information from other people, and (3) inferences from existing beliefs (Fishbein and Ajzen 1975). Not all information sources are equally influential. Information from direct experience leads to the strongest and most certain beliefs (Fazio 1989; Mathieson and Ryan 1991). Beliefs held with certainty are more difficult to change than beliefs that are less certain (Crocker, Fiske and Taylor 1984).

Some beliefs capture perceptions of causality, such as "a positive evaluation of a system will increase the probability of adoption." These beliefs form "implicit theories" for the individual (Markus and Zajone 1985). Hufnagel (1990) has shown that beliefs about the causes of events affect IS evaluations.

Raters report evaluations to the "sponsor," the person or group who will use the information. For example, the sponsor might be a developer constructing a new IS or a manager choosing a word processing package. The utility of an evaluation depends on the sponsor's objectives. For instance, if the sponsor wants to encourage adoption of the IS, he or she may want evaluations to be as positive as possible. However, the focus of this paper is on situations where evaluations provide information for decisions about MIS development activities. We assume that a useful evaluation is one that accurately reflects the extent to which an IS improves the performance of its users on a set of job-related tasks. If an IS does not achieve this objective, then a useful evaluation will say that the IS is poor. Although one might argue for other criteria, this paper emphasizes evaluation accuracy as the primary objective.

2.2 Model Overview

Figure 1 shows an overview of the model. Evaluations represent a comparison between normative beliefs and descriptive beliefs. Normative beliefs are the rater's opinions about attributes that the IS should possess. Descriptive beliefs are the rater's opinions about attributes that the IS does possess. The rater notes discrepancies between the two and forms an evaluation. The evaluation might be adjusted to serve the rater's interests. Finally, the rater reports the adjusted evaluation to the sponsor.
Figure 2. The Model in Detail
We do not suggest that these activities take place in a linear sequence. In fact, normative belief selection, descriptive belief formation, and belief comparison probably take place in a continuous cycle. This sequence will be discussed in more detail later.

A central tenet of the model is that raters are intelligent, goal-directed individuals, for whom IS evaluation is one of many responsibilities. This is consistent with recent theories of human behavior, including the DeNisi, Cafery and Meglino (1984) and the Landy and Farr (1980) models of employee performance appraisal, Porac's (1987) model of the cognitive processes used in questionnaire response, Ford and Ford's (1987) living systems framework, Graesser and Murachver's (1985) model of question answering procedures, and Shavitt's (1989) work on the functions of attitudes.

The next section discusses the goals that raters might pursue. A description of each of the model's components follows. Note that goal formation is outside the boundary of the current model. However, some ideas about the origins of goals are discussed later.

2.3 Goals

Evaluations are influenced by the purpose for which they are carried out (DeNisi, Cafery and Meglino 1984). Raters can have goals for both the outcome and the process of evaluation. Outcome goals include:

1. Accurately judge the system's support of some task set (task support goal).
2. Establish identity with various groups (social adjustment goal).
3. Express the rater's values (value expressive goal).

Consider each of these in turn. First, variations in task support goals can affect evaluations. Suppose rater R1 is interested in the system's support of task T1, while rater R2 is interested in task T2. If the IS supports T1 better than T2, then R1 should evaluate it more positively than R2. Miller (1989) has found support for the idea that task/system fit influences evaluations.

Social adjustment and value expressive goals can lead to evaluation biases. A rater who wants to be perceived as pro-technology may evaluate an IS positively, even if it is ineffective. The relationship between the rater and the designer can also be a source of bias. If raters view designers as having significant expertise and high status, raters might feel some pressure to agree with them.

Finally, the protective function might cause raters to bias their evaluations. For example, someone who feels threatened by an IS (perhaps because it automates a valued skill) may be inclined to reject it.

Since raters have several goals at the same time (Pervin 1989), their relative importance should be considered. For example, although a rater wants a useful IS, the goal of being perceived as a dynamic employee might be more important. The rater might be unwilling to voice doubts that could threaten this image.

Besides having goals for the outcome of the evaluation process, raters can have goals for the process itself. Perhaps the most important is the amount of effort a rater is willing to spend. The higher the rater's interest in the outcome, the more motivated the rater might be to spend time evaluating the IS (Abelson and Levi 1985).

Process and outcome goals are not independent, since processes influence outcomes. For example, suppose a rater wants to obtain an effective IS. His or her implicit theory of development indicates that excessive criticism of an early prototype will lead to conflict, reducing the developer's involvement with the project, and thereby reducing the chances of obtaining a good IS. The rater therefore might be less critical of an initial prototype than is justified, to avoid creating excessive conflict with the developer.

The same distortion might occur for different reasons. For example, another rater might limit criticism of a prototype because he or she is not interested in the IS and does not want to spend time evaluating it. This rater's implicit theory says that mild criticisms can be easily addressed and his or her input will no longer be required. Therefore, the rater might be less critical of a prototype than is justified. Both raters distort their evaluations in the same way, but because of different goals.

2.4 Normative Beliefs

We now turn to the components of the model, shown in Figure 2. In the rest of the paper, numbers in square brackets (e.g., [1]) refer to the components of Figure 2.

Normative beliefs [7] are the rater's opinions about the attributes that the IS should possess. At least three types of normative beliefs are likely to be important. First, task support beliefs identify specific features needed to perform job-related tasks. These beliefs could be based on ideas about how tasks should be performed. Second, there are normative beliefs that are not specific to a particular task, but reflect characteristics of systems that are normally desirable. For example, a rater might believe that menus are always preferable to command line interfaces. Third, normative beliefs might reflect environmental constraints, such as "cost less than $500 per workstation" and "run on IBM compatibles."
The normative beliefs that a rater applies in a specific situation depend on task support goals [1], beliefs about the support of specific tasks [3], and general beliefs about systems [4]. Task support goals help the rater select which set of task support beliefs apply. For example, a rater evaluating a word processor for the task "write a research paper" will look for a different set of features from the same rater evaluating the same system for the task of "preparing form letters."

The content of the normative beliefs depends on the rater's beliefs about the support needed for different tasks. Even if two raters agree on the task, they may not agree on the best way to support it. For example, one rater might feel that text formatting is best done with combinations of control keys, while another prefers using a mouse.

Normative beliefs [7] are also influenced by general beliefs about information systems. For example, a rater might believe that all systems should be fast. This normative belief applies regardless of the task.

Examining the system [5] can suggest options that the rater had not considered, influencing task-specific [3] and general beliefs [4]. For example, a rater might not have included support of sensitivity analyses in his or her normative beliefs for a decision support system. Perhaps their expense prohibits their use when the task is performed manually. When examining a prototype, however, the rater might realize that such analyses are economically feasible when computerized, and alter his or her normative beliefs. Thus, as raters gain experience, their expectations can change (Doll and Ahmed 1983). The less the user is familiar with the capabilities of IS technology, the more likely such changes may be.

Finally, the evaluation instrument [9] can affect normative beliefs. When an evaluation instrument asks raters to describe an IS on a particular dimension (such as reliability), raters might infer that the dimension is important (DeNisi, Caferty and Meglino 1984). Instrument effects might be particularly strong if (1) the rater is uncertain about his or her normative beliefs, as might be the case if the rater is a novice, and (2) the evaluation instrument (or the individual administering it) is highly credible.

2.5 Descriptive Beliefs

Descriptive beliefs [8] are the rater's opinions about the attributes that the system does possess. Figure 2 shows that descriptive beliefs are determined by the rater's experience with the system [5], which is in turn affected by the rater's normative beliefs [7], the system itself [2], and the rater's process goals [6].

Normative beliefs guide the raters' experience by suggesting what they should pay attention to. This is consistent with schema theory from cognitive psychology (Markus and Zajone 1985). For example, a rater evaluating a word processor might use normative beliefs related to the "write research papers" task, gathering information on the program's ability to handle tables, graphics, and mathematical symbols.

The tendency of normative beliefs to direct attention could limit the conclusions that a rater can draw. Suppose a rater evaluated a word processor's support for writing research papers. Later, he or she is asked about the system's suitability for preparing form letters. The rater may be unable to give an accurate opinion, since he or she did not pay attention to the system's attributes that are important for the this task.

Descriptive beliefs reflect the system that the rater sees. Obviously, different systems will be described by different beliefs. However, the nature of the experience raters have with the system will affect the descriptive beliefs they form. Two empirical studies are relevant here. First, Mathieson and Ryan (1991) found that direct experience with a system (that is, actually using it to complete a task) had a stronger effect on perceptions of task difficulty than indirect experience (e.g., watching another user solve a task, hearing the opinions of others, or drawing inferences from experience with other systems). Direct experience amplified raters' perceptions, but did not change their direction. The same effect might apply to perceptions of systems as well as perceptions of tasks. Second, Nosek and Ahrens (1986) found that a menu-based representation of a system resulted in more complete evaluations than a representation based on data flow diagrams. The menu-based representation may have allowed raters to more easily construct descriptive beliefs that were relevant to the evaluation.

The completeness of descriptive beliefs (i.e., their number and accuracy) depends on the amount of information the rater seeks. Two factors might restrict information search. First, there are limits on the rater's ability to process information (Abelson and Levi 1985). The greater the amount of data available and the greater the time pressure, the more selective the rater will be. Second, process goals could influence the rater's willingness to exert effort during the evaluation. Rater motivation has a strong impact on information gathering (Abelson and Levi 1985). The more important the outcome of the project is to the rater, the more carefully the evaluation will be considered.

2.6 Comparison

In the comparison stage [10], the rater compares normative and descriptive beliefs, and produces an evaluation. However, some important information may not be available. For example, a task novice with access to a complete system may have well-developed descriptive beliefs but incomplete normative beliefs. A task expert with well-developed normative beliefs may have incomplete
2.8 Evaluation Sequence

No particular temporal sequence has been suggested for evaluation activities. A reasonable sequence is that the rater would (1) choose an attribute that a normative belief suggests is important, (2) examine the IS to determine the system's value on that dimension, (3) compare the normative and descriptive beliefs, (4) decide whether to continue the evaluation, (5) choose the next normative belief, and so on. If goals suggest a particular outcome, then the rater might monitor the entire process to make sure that the evaluation is moving in the appropriate direction.

Of course, this is just one possible sequence. Individual behavior often does not follow completely predictable patterns. However, the sequence is reasonable, and captures the basic ingredients of the model.

3. TESTING THE MODEL

Some of the basic propositions of the model have been tested. This section examines empirical evidence supporting the model and makes suggestions for future tests. Given the critical role that goals play in the model, suggestions for operationalizing goals are given.

3.1 Current Evidence

Mathieson, Olfman and Ryan (1991) tested the idea that task support goals influence how raters examine systems, thereby affecting evaluations. This proposition is central to the model. Raters were shown one of two systems, each consisting of three subsystems. Each subsystem supported one task. The effectiveness of the subsystems varied within and between systems. The raters were also given a task support goal that asked them to either (1) focus on the system’s support of one task, or (2) simply “evaluate the system,” with no specific task being mentioned. Mathieson, Olfman and Ryan measured (1) the raters’ behavior (time spent and commands issued in each subsystem), (2) their overall evaluations of the systems, and (3) their evaluations of each subsystem.

The results supported the model. First, behavior was affected by task support goals. Raters examined subsystems relevant to their goals more closely than other subsystems. Second, overall evaluations depended on which subsystems the raters tested. Raters using the same system evaluated it differently, depending on the effectiveness of the parts of the system they examined. Third, raters were not able to clearly differentiate between the effective and ineffective parts of the system in their subsystem evaluations. They seemed to infer their subsystem evaluations from their overall evaluations, rather than the reverse.

3.2 Future Tests

While these results provide some empirical support for the model, more work is needed to fully test it. First, Mathieson, Olfman and Ryan did not examine the mediating role of beliefs. An analysis of the questions raters ask (as used in Lauer and Peacock 1990) could identify normative beliefs. These beliefs should predict raters’ evaluation
behavior, as measured by protocol analysis (as used in Card, Moran and Newell 1983) and logs of rater activity.

Second, it was suggested earlier that instruments can affect evaluations, especially for novices, by changing normative beliefs. This could be tested experimentally by designing instruments and systems so that (1) different instruments implied either positive or negative evaluations of the same system and (2) different systems were evaluated positively or negatively with the same instrument. The interaction of instrument and system should affect novice raters' overall evaluations. According to the model, novice raters' normative beliefs should also be influenced by the instrument. This could be tested by asking raters to list the features that an effective system should have, both before and after an evaluation. Some of the differences between the two lists should correspond to items on the evaluation instrument.

Third, the effects of goals besides task support should be examined. For example, social pressure could be manipulated by having raters deliver evaluations to people whose opinions about the system are known. If raters are motivated to comply, they might bias their ratings in the direction of the others' opinions. Motivation to comply can be manipulated by, for example, having raters either (1) deliver anonymous evaluations in writing or (2) meet face-to-face with people who know the identity of each rater.

3.3 Operationalizing Goals

Given the critical role of goals, their manipulation and/or measurement is important in testing the model. Mathieson, Offman and Ryan manipulated task support goals through instructions to subjects. They could be measured by asking raters what tasks they will be considering when evaluating the system. One problem is that task support goals can change as the rater learns more about the system's capabilities.

Social adjustment goals could be manipulated as discussed above. They could be measured in at least two ways. First, Snyder and DeBono (1989) discuss the construct of self-monitoring, representing the extent to which an individual will adjust his or her attitudes to gain social approval. Self-monitoring is an individual trait, applying across situations. Standard instruments for self-monitoring are available. The second approach is more situation specific. The theory of planned behavior (Ajzen 1989) predicts an individual's intention to behave in a certain way from, among other things, (1) the normative beliefs of referent others and (2) the motivation to comply with the referents. Relevant others, normative beliefs, and motivation to comply are measured separately in each situation, using procedures discussed by Ajzen and Fishbein (1980).

Values can be measured through general computer attitude instruments, such as that employed by Lee (1970) and later by Morrison (1983). Snyder and DeBono suggest that the strength of value expressive goals, that is, the extent to which a rater wants to express attitudes consistent with values, is related to self-monitoring. According to Snyder and DeBono, low self-monitoring is associated with high value expressiveness.

Protective goals are more difficult to handle. It may be unreasonable to expect accurate responses to direct questions, such as "Do you feel threatened by this system?" Herek (1987) developed a procedure for identifying attitudes that serve defensive purposes, but it is unlikely to be useful here, since it is not situation specific. For instance, Herek's Attitude Functions Inventory identifies the functions of attitudes towards stigmatized social groups, such as AIDS victims. An individual's attitude towards an AIDS victims is assumed to serve the same function across many situations. Protective goals, on the other hand, are likely to be very situation specific. For example, a rater might be threatened by a system that automates a task he or she performs, but not by a system that automates a task he or she used to perform in a previous job.

A wide range of protective reactions would be difficult to create in the laboratory, leaving aside the ethical questions about the level of psychological discomfort such a manipulation would create. It would be feasible, however, to examine situations in the field that can be assumed to encourage the development of protective goals.

Finally, an important process goal is the amount of effort that the rater is willing to spend in evaluating the system. This could be manipulated by altering the rewards attached to evaluation activities. If a rater is rewarded for performing a task well, he or she should be motivated to evaluate a system that could increase performance, if there is a chance that an evaluation will lead to adoption.

4. IMPLICATIONS FOR IS DEVELOPMENT

The model has many implications for IS development, although in the absence of empirical verification they are tentative. First, when sponsoring an evaluation, developers should consider their own objectives. For instance, a developer wanting to increase usage of an IS might want raters to evaluate the system positively. He or she might (1) encourage raters to focus only on tasks that are supported well and (2) use an evaluation instrument that directs raters' attention to the positive aspects of the IS. These actions should bias the evaluation in favor of the system.

Assume that the developer's objective is to obtain accurate evaluations of the system's support of a set of tasks. In other words, the developer wants raters to identify the system's shortcomings as well as its strengths. The model suggests that the developer should consider (1) who will be asked to do the evaluation, (2) what IS or IS description
they will evaluate, and (3) how the evaluation will be administered.

Choosing the raters is critical. First, raters should be motivated to provide an accurate evaluation of task support. This is probably most easily achieved if the tasks are personally relevant to the rater. Second, raters should have a fairly complete set of normative beliefs, that is, they should know what features a useful system would have. Third, the raters should have the time to devote to the evaluation. Fourth, the raters should not be unduly biased by social adjustment, value expressive, or protective goals.

These objectives may conflict. For example, task experts might have well-developed normative beliefs, but also may tend to value their expertise highly. They may want to protect themselves against an IS that could threaten their status. Further, task experts are often in high demand. They may have little time to devote to system evaluation (Gane 1989).

The second set of issues relates to the object being evaluated. The closer the object is to the target IS, the more accurate the descriptive beliefs (and hence the evaluation). However, constructing a realistic prototype can be expensive. One suggestion is to match the prototype to the raters’ expertise. If the raters have relatively high levels of IS and task expertise, they may need less information than novices to create an accurate mental model of the system.

Finally, there are issues related to the design of the evaluation session. Raters can be asked to focus on a particular set of tasks. If no task instructions are given, raters may focus on tasks that are particularly important to them. This will often be appropriate, but it is possible that some raters will be interested in tasks that the system was not designed to support.

Developers should make clear how the evaluations will be used. If raters know that their evaluations will affect their working lives, they might be willing to devote effort to the process. Further, if they know that their opinions will be confidential, raters’ social adjustment goals might not interfere. The disadvantage of explaining how evaluations will be used is that it might encourage protective distortions. Suppose raters are asked to evaluate an IS that will automate part of their job. Perhaps the IS will eliminate a task they find enjoyable. They might therefore negatively bias their evaluations in the hope of delaying the system’s implementation. Finally, developers should allow sufficient time for the evaluation.

We developed the model primarily to help design evaluation techniques that would assist developers. However, the model could also help researchers interested in using evaluations as a dependent variable. The model identifies variables that could confound research findings. For example, social adjustment goals might cause raters to evaluate the system in the way they think the researcher wants them to. Researchers using evaluations should consider the model’s implications for their study.

5. CONCLUSIONS

This paper describes a model of the process by which individuals evaluate IS. Raters are assumed to be intelligent and goal-directed, able to predict the consequences of their actions. The model described in this paper could help researchers and practitioners (1) better understand the psychological process involved in evaluation and (2) design effective evaluation procedures and instruments.

Many research challenges remain. The model must be tested and, when found wanting, modified. Two other issues are particularly interesting. The first is the manner in which raters choose goals. This was deliberately placed outside the boundary of the model, as shown in Figure 1, for two reasons. First, goal choice mechanisms are very complex and not well understood (Pervin 1989). Including them would have greatly increased the complexity of the model, making it difficult to understand and almost impossible to test. Second, the model is useful enough without explicitly considering goal choice. If goals can be measured, their effects can be predicted using the model. It was suggested above that goals can be measured fairly easily, with the possible exception of protective goals. If this is true, it is not necessary to explain goal choice within the model itself.

Nevertheless, goal choice is interesting because of the importance of goals in the model. Goal choice may provide a mechanism for linking organizational-level and individual-level phenomena. For instance, organizational design determines how tasks are allocated across employees (Rousselau 1978). The allocation of different tasks to different individuals will create differences in raters’ task support goals. Task allocation can also explain variance in motivation, which in turn affects process goals. For example, one buyer for a chemical firm might focus on commodities, while another concentrates on industrial equipment. Both might be asked to evaluate an IS that forecasts changes in commodities prices. Their motivation to provide an accurate evaluation could be very different.

Many other structural variables besides the rater’s organizational position could influence evaluations by affecting goals. The groups to which a rater belongs will determine his or her social adjustment goals. Protective goals could also be affected, leading to both positive and negative biases. Negative biases caused by protective goals are easily understood. For example, a rater could react negatively to an IS that could reduce his or her organizational power. However, protective goals could also cause a rater to positively bias an evaluation. Suppose a manager initiated a prototyping project. Prototyping lets users and developers test ideas fairly cheaply (Naumann and Jenkins 1982). If they decide that a cost effective system cannot be
developed, they should abandon the project. However, if the manager felt that doing so would threaten his or her credibility, he or she might positively bias evaluations to preserve the appearance of success.

The second major research issue is the role of individual differences. Individual difference variables could moderate some of the effects in the model. For example, social approval is more important for some people than for others (Snyder and DeBono 1989), moderating the impact of social adjustment goals on evaluation adjustment.

Another example is IS expertise. Novices often rely on memorized sequences of commands (called rules) to perform system-related tasks (Carroll, Olson and Anderson 1987). They have no understanding of how a system works. If it changes, they cannot function until they learn new rules. Experts, however, develop sophisticated mental models of systems that allow them to more easily adapt to change. An effect of this may be that someone relying on rules cannot accurately evaluate a new system that is too different from the current one. He or she may reject the new system because his or her existing rules do not apply. However, a more expert rater might understand the new system and its implications for task performance, and evaluate it accurately.

To summarize, this paper presented a model of the IS evaluation process. Raters are intelligent, goal-directed individuals, who can anticipate the consequences of their actions. A number of ways in which evaluations could be biased were identified. Suggestions were made for improving current practice and future research.

6. REFERENCES


7. ENDNOTE

1. This list is based on Shavitt (1989, pp. 312-313).