

Relative importance of evaluation criteria for enterprise systems: a conjoint study

Mark Keil* & Amrit Tiwana†

*Georgia State University, Department of Computer Information Systems, J Mack Robinson College of Business, 35 Broad Street, Room 422, Atlanta, GA 30303, USA, email: mkeil@gsu.edu, and †Iowa State University, College of Business, 2340 Gerding Business Building, Ames, IA 50011-1350, USA, email: tiwana@iastate.edu

Abstract. *While a large body of research exists on the development and implementation of software, organizations are increasingly acquiring enterprise software packages [e.g. enterprise resource planning (ERP) systems] instead of custom developing their own software applications. To be competitive in the marketplace, software package development firms must manage the three-pronged trade-off between cost, quality, functionality. Surprisingly, prior research has made little attempt to investigate the characteristics of packaged software that influence management information system (MIS) managers' likelihood of recommending purchase. As a result, both the criteria by which MIS managers evaluate prospective packaged systems and the attributes that lead to commercially competitive ERP software products are poorly understood. This paper examines this understudied issue through a conjoint study. We focus on ERP systems, which are among the largest and most complex packaged systems that are purchased by organizations. In a conjoint study, 1008 evaluation decisions based on hypothetical ERP software package profiles were completed by managers in 126 organizations. The study represents the first empirical investigation of the relative importance that managers ascribe to various factors that are believed to be important in evaluating packaged software. The results provide important insights for both organizations that acquire such systems and those that develop them. The results show that functionality, reliability, cost, ease of use and ease of customization are judged to be important criteria, while ease of implementation and vendor reputation were not found to be significant. Functionality and reliability were found to be the most heavily weighted factors. We conclude the paper with a detailed discussion of the results and their implications for software acquisition and development practice.*

Keywords: software selection, packaged software development, ERP systems, attributes, software design trade-offs, enterprise systems

1 INTRODUCTION

Packaged software applications now represent a \$183 billion annual worldwide market (Software & Information Industry Association, 2003). Although the market for packaged software originated with standalone, function-specific applications, it has matured to encompass complex enterprise-level software. Packaged software is estimated to represent nearly 70% of corporate business software expenditures (Holland & Light, 1999). Worldwide sales of enterprise resource planning (ERP) packages in the manufacturing sector alone reached \$8.9 billion in 2002 (ARC Advisory Group, Inc., 2003). Software packages like these offer organizations an alternative to internally developing such enterprise software systems. In this paper, we do not intend to take sides in the debate of whether or not enterprise software packages are 'good' or 'bad' for organizations. Whether such systems are 'good' or 'bad' and whether they are 'flexible' or 'inflexible' is not our focus. Instead, we offer research that can help guide those organizations that are faced with the prospect of acquiring such systems and who seek a set of weighted criteria by which to evaluate alternative packages. 2

While buying and implementing packaged enterprise software applications may appear risk-free, the majority of such efforts fail. In a survey of 232 managers conducted by Robbins-Gioia (a Virginia based consulting firm), 51% viewed their ERP implementation as unsuccessful (IT-Cortex, 2003). The complexity of integrating and implementing complex packaged software carries some of the risks typically associated with custom-developed software. The attributes that affect purchase acquisition likelihood clearly extend beyond the usual concerns of cost, schedule and functionality into factors that influence implementation, customization and integration of the software with other systems. 3

Unfortunately, most organizations lack a rigorous software procurement process, which makes it difficult to make effective software purchasing decisions (Shang & Seddon, 2002; Appel *et al.*, 2003). By the same token, the lack of an understanding about how buyers evaluate such software can lead packaged software firms to either under-invest in software attributes that influence customers' purchase decisions or to over-invest in attributes that do not have much influence. Unfortunately, no systematic research exists on what attributes of enterprise systems (such as ERP software) affect managers' likelihood of recommending purchase. Without knowledge about such criteria: (1) organizations risk making inappropriate acquisition decisions; and (2) packaged software development firms risk making uninformed assumptions about what attributes the market truly values. 4

In this study we surveyed 126 management information system (MIS) managers to gain insight into the factors that influence enterprise software package selection. Our aim was to address two basic research questions.

- 1 What are the salient criteria for evaluating packaged enterprise-level software?
- 2 What are the relative weights that managers ascribe to these criteria?

To address the first question, we conducted a literature review aimed at identifying criteria that have been suggested for evaluating packaged software. To address the second question,

which is what most distinguishes this study from prior research, we conducted a conjoint study to gain insight into the relative weights that managers ascribe to these criteria. The remainder of the paper is organized as follows. In the next section, the relevant literature on packaged software selection is discussed. In section 3, we develop a formal model for predicting managers' likelihood of recommending purchase. In section 4, we describe the data collection methodology and analyses and in section 5 we present our results. Section 6 discusses the implications of our findings for packaged software design and for acquisition, followed by the key conclusions of the study.

2 PRIOR WORK

Prior research has suggested that there are significant differences between the *package* development environment (in which software is developed as a product for external sale) and the *custom* environment (in which software is either developed in-house or under contract and is intended for internal use) (Grudin, 1991; Keil & Carmel, 1995; Sawyer, 2001). In terms of the software acquisition process, the custom environment allows the customer to specify his/her requirements prior to development and the software is then developed to meet those needs. In the package environment, software acquisition is fundamentally different in that the customer must select from an array of packages which have already been developed with the aim of finding one that most closely meets his/her needs. Sawyer (2001) observes that the acquisition of enterprise systems involves a significant investment and is likely to attract senior management's attention.

Acquisition of complex packaged software such as ERP systems often involves making trade-offs among a variety of attributes such as cost, degree to which the software meets the organization's requirements, customizability and so forth. A classic dilemma that frequently emerges is whether the organization should adapt its business processes to fit the software or whether the organization should adapt the software to fit existing business processes. Adapting organizational processes often means massive organizational change, thus raising significant implementation challenges. Going the other route of customizing packages once they are purchased can prove to be a costly process. In most cases, the acquisition of enterprise packages involves some 'mutual adaptation of new software-based processes to existing organizational processes' (Sawyer, 2001, p. 100). Typically, this means that the package not only requires integration with other systems, but also some customization to the idiosyncratic business processes of the adopting organization (Holland & Light, 1999; Murphy & Simon, 2002; Stensrud & Myrtveit, 2003). Thus, there are many factors that need to be considered in purchasing enterprise packages. However, no rigorous research exists on the criteria that enter into the evaluation process, and the relative importance of such criteria.

Most of the information available on selection criteria and the process of acquiring a software package comes in the form of practitioner-oriented, prescriptive publications (e.g. Kintisch, 1987). We first review a representative sampling of these and then develop an empirical model based on the salient attributes that emerge from the review.

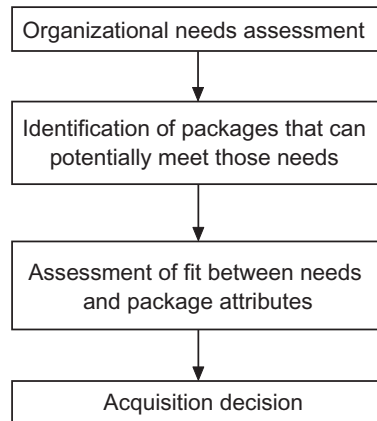


Figure 1. General process framework for choosing packaged software.

As illustrated in Figure 1, the task of choosing packaged software begins with an assessment of the organization's business needs. This is typically followed by an investigation of available software packages that might meet those needs. The list of potential packages is then narrowed to a manageable few, which are then subjected to further scrutiny, until a choice is made. In many cases, a request for proposal (RFP) is generated based on the needs assessment and this document is sent out to software vendors, leading to a set of proposals which are then evaluated. Typically, a 'gap-fit' analysis is performed whereby a package's features are mapped to an organization's requirements (Sawyer, 2001).

Two broad categories of risk must be considered in the software acquisition process. The first involves attributes of the software package itself. Issues of functionality provided by the software and the software's reliability are examples of risks that fall within this category. Software that lacks the appropriate functionality carries the risk of failing to meet user requirements (Sherer, 1993). Such risk is heightened when managers rely heavily on vendor-supplied marketing material instead of conducting a detailed requirements analysis. The second broad category of risk involves the implementation of the software. This type of risk arises from difficulties in modifying the software or business processes to fit each other and integrating the package with existing systems (Henfridsson, 2000). To minimize these types of risk, managers need to establish appropriate selection criteria to help them evaluate which enterprise software package will best suit the needs of their organization. In the remainder of this section, we review the literature on packaged software selection criteria.

Very little has been written about packaged software selection criteria in academic journals. We therefore conducted a literature search beginning with academic papers and then expanding it to practitioner articles. Our objective was to identify prior research in leading information systems and computer science journals that had specifically studied the criteria that managers use in evaluating packaged enterprise software systems. We used the ABI INFORM database to conduct the literature review. We first searched using two sets of keywords: (1)

enterprise systems or ERP systems; and (2) selection criteria. We focused the preliminary search on peer-reviewed academic articles on the topic, examining the major journals in information systems and computer science. The former included *Information Systems Journal*, *MIS Quarterly*, *Journal of Management Information Systems*, *Information Systems Research*, *European Journal of Information Systems*, *Management Science*, *Decision Sciences*, *Communications of the ACM* and *Decision Support Systems*. The latter set of journals included *IEEE Transactions on Software Engineering*, *IEEE Transactions on Engineering Management*, *IEEE Software*, *IEEE Computer*, *ACM Transactions on Information Systems*. In our review of the literature, we found only three scholarly publications on the subject (Chau, 1995; Montazemi *et al.*, 1996; Bernroider & Koch, 2001). Because there were relatively few scholarly articles on the subject, we also surveyed the literature for a representative set of practitioner-oriented articles on the subject of packaged software selection. We used a similar set of keywords for this search but did not narrow down this pool of publications to a specific set. To further broaden this pool, we also searched teaching cases and case videos on the topic, using Harvard Business School's teaching case repository. The resulting articles therefore provided a representative sampling of the various criteria that managers use in evaluating enterprise software packages. Table 1 lists the articles that we identified and provides some brief descriptive information on each.

After reviewing each article, we looked for patterns that would indicate those attributes around which there seemed to be some consensus. If more than one article mentioned the importance of a particular attribute, we made note of this. In synthesizing the literature, we identified seven factors that seemed to represent a consensus view of items that should be considered in evaluating packaged software. These included: cost, reliability, ease of use, ease of customization, ease of implementation and vendor reputation.¹ Table 1 shows how the articles we reviewed mapped against each of these factors. Theoretically, the package assessment process is a multi-attribute decision-making process and the manner in which managers simultaneously use these attributes to arrive at an overall assessment of the system provides insights into the relative importance of various software attributes. Below, we discuss each factor that we identified.

2.1 Cost

Cost is an obvious consideration for any purchase, and it stands to reason that packaged software is no exception to this rule. Seven of the 10 sources we examined mentioned cost as an important factor in selecting software packages. Goldenberg (1991) stresses the importance of understanding not only what the price is, but what the price covers. For example, does the price include training, support, or upgrades? This is important, as the cost of implementing a package can often exceed the cost of the buying the package. Bryce & Bryce (1987) stress the importance of knowing whether there is any maintenance or other ongoing costs to use the

¹We make no claim here that the list of factors is exhaustive, only that it represents a reasonable subset of the numerous factors that have been mentioned in the literature.

Table 1. Representative sampling of articles on packaged software

Publication reference	Type of publication	Type of packaged software	Context and type of study
Johannsen, 1980	Practitioner article	Not specified	Informational article that outlines software selection criteria
Brown & Stephenson, 1981	Practitioner article	Not specified	Informational article that describes pros and cons of packaged software and describes general factors that should be considered in evaluating software packages
Bryce & Bryce, 1987	Practitioner article	Not specified	Informational article that tackles the question of make vs. buy and describes how to evaluate a package
Pivnicny & Carmody, 1989	Practitioner article	Hospital information system	Description of the RFP process used by a 365-bed hospital to update its information system
Keller, 1994	Practitioner article	Manufacturing applications	Informational article describing the challenges of selecting a package and the importance of choosing from a vendor that will be around for foreseeable future
Goldenberg, 1991	Practitioner article	Sales and marketing software	Short informational article describing three factors that should be considered when choosing a package: vendor reputation, software functionality and pricing
Montazemi <i>et al.</i> , 1996	Academic article	29 microcomputer applications and one mainframe software package	Perceptions of information centre product specialists (ICPSs) and end users were compared with regard to the perceived ease of use (EOU) and usefulness of 30 different software packages. End-users perceived the software packages to be less useful than the ICPSs, suggesting that the perceived value of a package's functionality is likely to vary depending on the job role of the evaluator. They found no differences in perceived EOU between the two groups

Chau, 1995	Academic article	Not specified. Respondents asked to provide information on 'the most recent packaged software purchase'.	Survey of 122 small manufacturing businesses in Hong Kong aimed at understanding the importance that owners and managers of small businesses place on various packaged software selection factors
Bernroider & Koch, 2001	Academic article	Enterprise resource planning (ERP) systems	Survey of 138 Austrian organizations aimed at understanding differences in the ERP selection process between small or medium and large sized organizations. Prior to conducting their survey, they identified 29 different ERP selection criteria using the Delphi approach, but they do not report the results of their Delphi study. Unfortunately, in reporting the results of their survey, they present only those criteria with a 'strong relationship to organization size' (Bernroider & Koch, 2001, p. 254).
Romanow <i>et al.</i> , 1998	Academic case	Enterprise software for managing parts distribution	Case study of Timberjack Parts and the process and criteria that they used to select packaged software

RFP request for proposal.

system, whether the vendor will charge for providing modifications, and whether there are other costs associated with installing the package. Johannsen (1980) distinguishes between operating and maintenance costs vs. purchase and installation costs. Chau (1995) found that cost was a significant factor for small manufacturing firms and that managers viewed cost as a more important factor than did the owners of the firms.

2.2 Reliability

Brown & Stephenson (1981) suggest that one of the advantages of buying packaged software is that it represents a superior product relative to what might be developed in house. Thus, what drives many packaged software purchases is the desire to have a higher quality, more reliable, piece of software. Bryce & Bryce (1987) stress the importance of the vendor's reputation for reliability. Johannsen (1980) lists reliability as one of 11 objectives that every software package should satisfy. 5

2.3 Functionality

Goldenberg (1991) suggests that buyers should carefully examine the 'sophistication of the functions included in the software'. Bryce & Bryce (1987) stress the importance of 'features' as one of several points to consider. Johannsen (1980) puts functionality first in a list of 11 objectives that every software package should satisfy. Pivnicny & Carmody (1989) list functionality as the first of nine criteria for evaluating packaged software. Montazemi *et al.* (1996) suggest that usefulness is an important factor for evaluating software packages. Usefulness is closely related to functionality and captures the relationship between functionality and the task at hand. 6

2.4 Ease of use

Bryce & Bryce (1987) stress that the software package should be 'easy and straightforward to use'. Johannsen (1980) puts 'usability' third in a list of 11 objectives that every software package should satisfy. Chau (1995) stresses the importance of 'ease of use/user-friendliness' as one of four 'software-technical side' factors that should be considered in selecting packaged software. Montazemi *et al.* (1996) suggest that ease of use is an important factor in evaluating software packages. 7

2.5 Ease of customization

Bryce & Bryce (1987) suggest that buyers consider 'what changes to the system are required' in order to meet requirements. Johannsen (1980) notes that flexibility is an important consideration in selecting packaged software. By this, he means whether the package can be 'easily changed and adapted'. Goldenberg (1991) also emphasizes the importance of being able to customize the software package. He suggests that the buyer even consider the possibility of 8

purchasing the source code where possible in order to have the ability to customize the software. Bernroider & Koch (2001) report that adaptability/flexibility of the package is an important factors and that smaller organizations put a higher value on this factor with 68% of respondents from small and medium companies rating this as 'very important' as compared with 50% of respondents from large companies who rated it as 'very important'.

2.6 Ease of implementation

Pivnicny & Carmody (1989) list 'application implementation and ongoing support' as one of nine criteria for evaluating packaged software. Bernroider & Koch (2001) found that time to implement is an important factor and present evidence that smaller organizations put a higher value on 'short implementation time', with 37% of small and medium companies rating this factor as 'very important' as compared with 30% of large companies who rated it as 'very important'. Romanow *et al.* (1998) note that the time and cost required to implement the package surfaced as a key factor in one company's packaged software selection process.

2.7 Vendor reputation

9 Brown & Stephenson (1981) list the 'supplier' as the first of six general factors that should be considered in evaluating packaged software. Pivnicny & Carmody (1994) suggest that vendor stability is a factor to consider. Chau (1995) cites 'reputation' as the first of four factors to consider under 'Vendor – non-technical side'. Goldenberg also recommends that buyers explicitly consider 'vendor reputation'. Keller (1994) even emphasizes the importance of vendor stability over functionality, stating that 'it is not so important that a company can meet a functional specification as it is that it has a viable plan to be in business in 5 years'. Bernroider & Koch (2001) present evidence that larger organizations put a higher value on 'market position of vendor', with 35% rating this factor as 'very important' as compared with 17% of small and medium companies who rated it as 'very important'.

To summarize, seven factors emerged from our review of the literature. These can be classified into two broad categories:

- 1 attributes of the software package: cost, functionality, reliability and ease of use; and
- 2 implementation attributes: ease of customization, ease of implementation and vendor reputation (under which support is subsumed).

These attributes along with their sources in prior work are summarized in Table 2.

In conducting a review of the literature, we were unable to find any articles – academic or practitioner-oriented – that discussed what the relative importance of various factors are or what weights they should be given in deciding which software package to purchase. In a Harvard Business School case, Romanow *et al.* (1998) describe Timberjack Corporation's selection criteria and present a set of weights associated with them, but in the accompanying case video Romanow (who participated in the selection process at Timberjack) admits that the weights they used were not particularly well thought-out:

Table 2. Selection criteria for packaged software

Software package attribute	Source in prior work				
	(Johannsen, 1980)	(Brown & Stephenson, 1981)	(Bryce & Bryce, 1987)	(Pivnicny & Carmody, 1989)	(Keller, 1994)
Cost	•	•	•	•	
Reliability	•	•	•		
Functionality	•	•	•	•	
Ease of use	•		•		
Ease of customization	•	•	•		
Ease of implementation				•	
Vendor reputation		•		•	•

If you're going through this process, spend a lot more time than we did on these weights. It's unfortunate, but we spent a fraction of the time that we should have establishing these weights and coming to a consensus. and certainly that became an issue (Keil, 1998).

Curiously, aside from the discussion in the Timberjack case and video, there appears to be no information in the literature regarding the relative weights that managers assign or should assign to various selection criteria.

Source in prior work					Variance in terminology
(Goldenberg, 1991)	(Montazemi <i>et al.</i> , 1996)	(Chau, 1995)	(Bernroider & Koch, 2001)	(Romanow <i>et al.</i> , 1998)	
.		.		.	Goldenberg (1991) uses the term 'pricing'
.	.			.	Package features (Brown & Stephenson, 1981)
	.	.		.	Software features (Bryce & Bryce, 1987)
.			.	.	Usability (Johannsen, 1980)
			.	.	Flexibility (Bryce & Bryce, 1987; Johannsen, 1980; Romanow <i>et al.</i> , 1998)
				.	Adaptability (Bernroider & Koch, 2001)
			.	.	Ability to customize (Goldenberg, 1991)
			.	.	Application implementation (Goldenberg, 1991). This is discussed implicitly as being an important factor in (Goldenberg, 1991)
.		.	.	.	Ongoing support & vendor stability (Goldenberg, 1991). Support (Romanow <i>et al.</i> , 1998)

3 MODEL SPECIFICATION

Because we are interested in assessing how the seven salient attributes of packaged enterprise software predict the managers' likelihood of recommending purchase, we formally specify the model below.

$$\begin{aligned}
 \text{Likelihood of recommending purchase} = & \alpha_0 + \alpha_{\text{functionality}} \times \text{functionality} + \alpha_{\text{reliability}} \times \text{reliability} \\
 & + \alpha_{\text{cost}} \times \text{cost} + \alpha_{\text{customization}} \times \text{ease_of_customization} \\
 & + \alpha_{\text{eou}} \times \text{ease_of_use} + \alpha_{\text{reputation}} \times \text{vendor_reputation} \\
 & + \alpha_{\text{implementation}} \times \text{ease_of_implementation} + \varepsilon_2 \quad (1)
 \end{aligned}$$

10

In each model, one of the two dependent variables of interest is expressed as a function of the seven attributes discussed in the preceding section. The alpha coefficients of Equation 1 provide insights into the relative importance of these attributes in explaining managers' likelihood of recommending purchase. The results of this model can provide important insights into the behavioural intentions of organizational buyers, providing the basis for constructing a 'balanced scorecard' that applies different weights to the various attributes and can be used to evaluate packaged software. The results can also inform developers of packaged software about the relative importance of attributes that can affect the commercial success of a packaged software product offering.

4 METHODOLOGY

A multi-method approach (Mingers, 2001) was used to conduct the study. This is summarized in Figure 2. We began with a qualitative assessment of patterns of evaluation criteria in a review of the literature on enterprise systems selection to develop a base conjoint profile (Phase 1; discussed in the preceding sections), followed by a quantitative conjoint study involving 126 managers (Phases 2 and 3), and finally a qualitative phase in which six managers were surveyed to help us interpret some of the less intuitive findings from the quantitative phases of the study (Phase 4).

A conjoint study was conducted to assess the decision criteria that MIS managers in 126 different US organizations use in making acquisition decisions about packaged enterprise software applications. The recommended guidelines for empirical research on software engineering phenomena were closely observed throughout the data collection process (Kitchenham *et al.*, 2002). We used a conjoint survey design, which is a multi-attribute judgement analysis technique involving *posterior* decomposition of a respondent's decision process (Priem, 1992). In this approach, the underlying structure of the respondents' decision rules over a series of decisions can be statistically inferred by estimating the regression equations underlying the model (Louviere, 1988). Each respondent evaluates a series of independent decision profiles which represent a software package with a different *bundle* of attributes. In this study, the profiles were enterprise software packages and the attributes were the seven criteria that we identified in our literature review. Each respondent was shown several software package profiles and each profile had a different combination of package attributes. For each profile, subjects were asked to provide an assessment of the likelihood of recommending that their organization acquire this package.

From these responses, the part worth utility for each package attribute was computed. This indicates the value that each respondent associates with a given package attribute. The

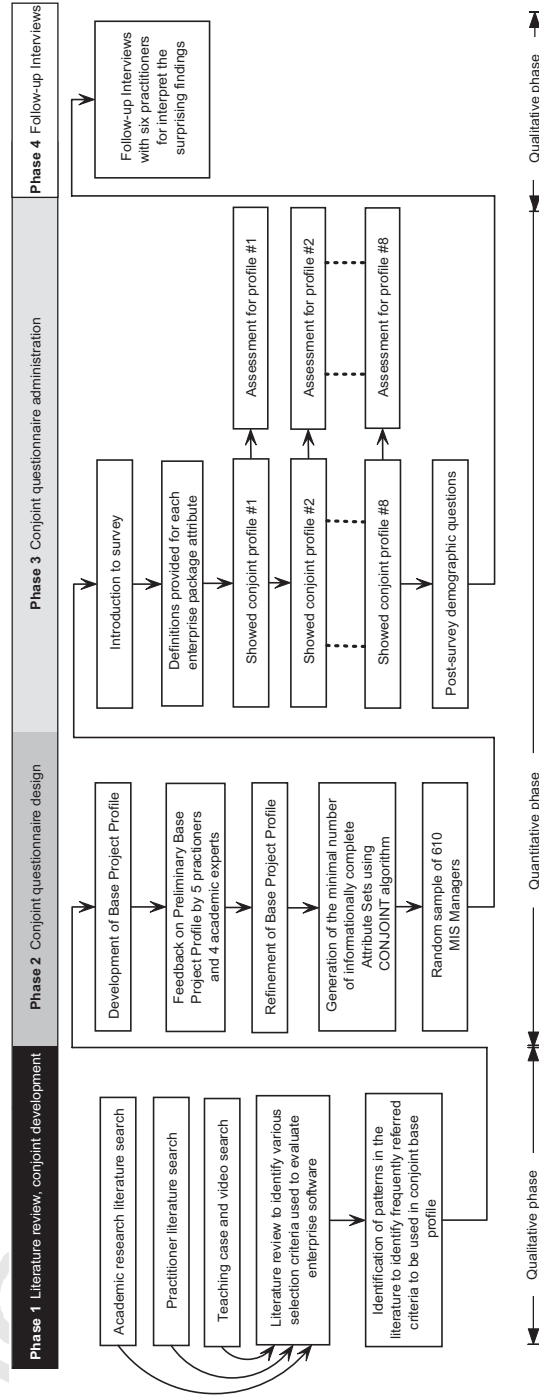


Figure 2. An overview of the multi-method research approach used in the study.

relative importance of each attribute is indicated by its regression weight, α_j . α_j , which represents the relative importance of the j^{th} attribute is estimated by considering it *in the context of the other six attributes* in the model. Overall likelihood of recommending purchase for a given enterprise system was then estimated as a multi-attribute function of the seven attributes described in (1) and summarized in (2).

$$\text{Likelihood_of_recommending_purchase} = \alpha_0 + \sum_{\text{package attribute } j=1}^7 \alpha_j \times X_j + \varepsilon_1 \quad (2)$$

Where, α_j = relative importance of package attribute j
 X_j = level of attribute j for a given ERP package
 ε_1 = residual error term

Our choice of the conjoint design approach was motivated by three considerations. First, a conjoint design allowed us to examine how managers concurrently consider all seven criteria in assessing each software package. Second, this approach combines the generalizability of a survey with the precision of a laboratory experiment. Third, conjoint analysis is frequently used by marketing researchers to understand the trade-offs that individuals make among product attributes when faced with purchase decisions.

As a complement to the conjoint study, we conducted a follow-up survey with a selected group of six MIS practitioners. The subjects who participated in the survey represented an opportunistic sample of industry practitioners who were chosen because of their experience in packaged software selection and their ability to reflect deeply on the findings that emerged from the conjoint study. This approach allowed us both to validate the results of the conjoint study and to gain additional insight into why practitioners might be inclined to put more emphasis on some attributes and less emphasis on others.

4.1 Conjoint questionnaire design

A fractional factorial conjoint design was implemented using SPSS CONJOINT algorithm, which generates the minimum number of attribute combinations needed for data collection. This algorithm generated eight software package profiles that were later evaluated by each respondent. The seven package attributes used in the study were: functionality, reliability, cost, ease of use, vendor reputation, ease of implementation and ease of customization. As part of the conjoint design, each respondent was presented with the same series of eight enterprise software package profiles, each of which described a software package in terms of the seven attributes discussed earlier (see Appendix). These criteria can be thought of as product attributes and their values were set to either 'high' or 'low'. For example, for a given software package profile, cost was set to 'high' or 'low'. Each scenario presented a different product profile with a unique combination of attribute values for the set of seven attributes.

The definition of each attribute was presented to each respondent at the beginning of the questionnaire (see Appendix). Functionality was defined as the 'extent to which the package contains all the features and functions specified in your RFP'. Reliability was defined as the

'overall quality and stability of the software'. Ease of use was defined as 'the intuitiveness of the user interface'. Vendor reputation was defined as 'the reputation that the vendor has established for unconditionally standing behind its products'. Ease of implementation was defined as 'the set-up time and effort that will be required to implement this package in your organization'. Finally, ease of customization was defined as 'the extent to which this package can be easily modified to meet your organization's unique needs.

The dependent variable was measured using a nine-point semantic differential scale and assessed managers' perceptions of the likelihood that they would recommend that their organization acquire the software package. Materials were pretested with a group of five mid-level MIS executives and four academic experts in software project management to ensure that the instrument was unambiguous, possessed face validity, and that the project profiles were realistic.

4.2 Survey sample and data collection

For the conjoint part of the study, Dun and Bradstreet's Million-Dollar Directory of executives was used to identify a random sample of *MIS Managers* in 610 organizations. MIS managers are mid-level to senior-level managers who generally report to the chief information officer (CIO) of an organization. MIS managers are frequently responsible for developing their organization's information technology portfolios. Because MIS managers are responsible for software purchasing decisions for their organizations (Appel *et al.*, 2003), they were the ideal respondent for our study. We contacted each manager through the mail, and provided a URL for the web-based version of the survey instrument in the cover letter. On the survey website, the respondents were provided instructions, the conjoint profiles and a short questionnaire. Following the preliminary instructions, each respondent was presented the first enterprise software package description, which they evaluated by rating the two dependent variables. Seven additional project profiles – each identical to the first but with different combinations of attribute values – were then evaluated by each respondent. Finally, each respondent completed a short questionnaire in which they answered demographic questions and a series of questions designed to measure the importance ascribed to vendor firm size, vendor stability, post-acquisition support and subjects' willingness to adopt a package that runs on a different platform than the one that is the dominant standard in their organization. Subjects were also asked to provide us with an indication of how much confidence they had in their own evaluations. 107 mailings were returned as undelivered and three additional executives declined to respond because of company policy. We received 126 *completed* sets of responses from managers in 126 organizations (25.2% response rate). This provided us 1008 package evaluations (7 software package profiles × 126 responding organizations) for the purpose of statistical analyses.

4.3 Respondent demographics

On average, the respondents had 16.3 years of MIS experience and had previously been involved in making software package selection decisions for 32 software packages. This sug-

gests that our respondents were experienced managers who were knowledgeable about software package selection. On average, the organizations in our sample had annual revenues of \$47.3 million. The sample represented a broad spectrum of firms engaged in both manufacturing and services.

5 RESULTS AND DISCUSSION

A multiple regression model was used to estimate the equation, based on the data from 1008 assessments by MIS managers in 126 organizations. The effects of the seven enterprise system attributes on managers' likelihood of recommending that their organization acquire the system are summarized in Table 3.² The *t*-values obtained for the standardized regression coefficients indicate that five of the seven attributes were statistically significant. Four of these exhibited *t*-values that were significant at the $P < 0.001$ value which indicates a 99% confidence level that the results obtained in the regression analyses are robust and not by chance.

The results show that functionality, reliability, cost, ease of customization and ease of use are all statistically significant factors that influence likelihood of recommending purchase. To control for alternative explanations, we also tested whether the size of the vendor organization, stability, after-sales support and compatibility with existing platforms influenced the likelihood of acquisition. None of these variables were statistically significant. This suggests the five aforementioned attributes are the key variables that managers consider in their software acquisition decisions for enterprise systems. The large amount of variance explained by these variables supports this conclusion. The five statistically significant software package attributes explained more than 50% of the variance in managers' likelihood of recommending purchase. This high

Table 3. The effects of each enterprise software package attribute on likelihood of recommending purchase

Enterprise system attributes	Effect on likelihood of recommending purchase Standardized alpha (<i>t</i> -value)
Functionality	0.457 20.03***
Reliability	0.464 (20.34)***
Cost	-0.253 (-11.08)***
Ease of customization	0.129 (5.67)***
Ease of use	0.073 (3.19)**
Vendor reputation	0.007 (0.29)
Ease of implementation	0.000 (0.01)
Model $R^2_{Adjusted}$	50.6%

** $P < 0.01$; *** $P < 0.001$.

²This study was part of a larger effort that also examined how these same enterprise system attributes influence perceived COTS value. The results of that analysis are reported in Keil & Tiwana (2005) and were quite consistent with the results obtained here for likelihood of recommending purchase.

level of variance explained by our models suggests that these five attributes are highly predictive of managers' perceptions about their likelihood to recommend purchase.

The absolute value of the path coefficients in Table 2 provides an indication of the relative weights that managers ascribe to each attribute. This can be seen graphically in Figure 3.

Functionality and reliability appear to be of paramount concern, with cost, ease of customization and ease of use being of somewhat lesser concern in managers' minds. Interestingly, neither vendor reputation nor ease of implementation was found to be significant in terms of likelihood of recommending purchase.

As a complement to the conjoint study results, we conducted a follow-up survey with a selected group of six MIS practitioners (Phase 4), five of whom provided detailed responses to the six questions that we posed. The practitioners were presented with the graph shown in Figure 3 and the first question that we asked was: 'Do you agree with the general pattern (relative importance) of results in the graph?' All of the MIS practitioners indicated that they agreed with the general pattern of the results, thus providing some validation for the results that were obtained in the conjoint analysis. One respondent wrote: 'I would agree with the general pattern'. Another respondent wrote: 'Having just gone through an ERP implementation with a client I can affirm that, given these five factors, they probably would have ranked them the same way'.

The second question we asked was whether the '5 factors in the graph accurately reflect the criteria that you would use in evaluating enterprise software packages?' and 'If not, what factors are missing in this framework?' Here, several suggestions were offered relating to factors that one might want to consider adding to those that we investigated. One respondent suggested adding 'maintenance cost, or annual operations cost' while another suggested 'fit of the package with the company's IT strategy'. A third respondent expressed a similar idea which he

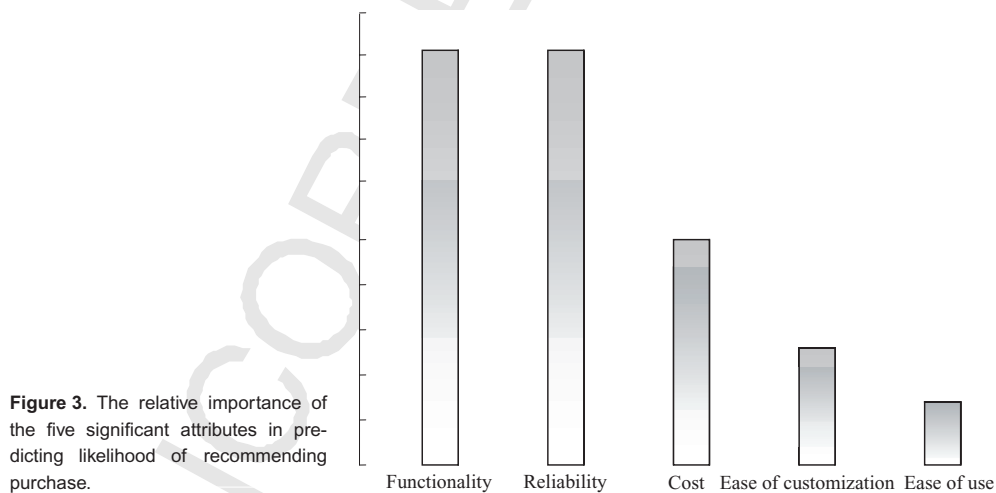


Figure 3. The relative importance of the five significant attributes in predicting likelihood of recommending purchase.

called 'cultural congruence' noting that 'some companies have gone under or at least lost millions of dollars in failed [ERP] implementations because for this reason'. His point was that a company that lacks any type of process discipline will run into problems trying to implement an enterprise package that depends on processes being defined in a certain way. This same respondent suggested adding 'integration with existing investments' to the evaluation criteria, noting that:

if the new package starts clean with everything new, it washes away years of effort and millions of dollars, along with pride of ownership of people who may be key to the success of the package . . . [Thus,] whether a package takes advantage of existing investments (for those components which the company wishes to retain and is not deliberately attempting to remove) can make or break the success of the package in the environment. Key resistors can be converted into champions or advocates with the acknowledgement of work already performed and serving the company.

Based on the comments received, it is clear that the seven factors we examined could be supplemented with additional factors. However, our purpose was not to test an exhaustive set of factors, but rather, to develop a parsimonious list that captured the essential criteria based on our review of the literature. Thus, while the list of factors could be expanded, we believe that there is value in examining these seven factors and the relative importance ascribed to the five that were found to be statistically significant in the conjoint analysis.

Figure 3 reveals that the most important factor that managers consider in evaluating enterprise packages is the functionality offered by the software package. That a high degree of importance is placed on functionality is perhaps not surprising. The emphasis on functionality was clearly articulated by one of the conjoint respondents: 'If the thing won't do what you want when you want, it's worthless'. This sentiment was echoed by the respondents to our follow-up survey, one of whom noted that he 'would weight functionality as at least twice as important as any of the other factors'.

The second most important factor is the software's reliability. In our follow-up survey, we asked MIS practitioners whether they thought that the high importance ascribed to reliability reflects dissatisfaction with the overall quality of packaged software available today or whether they thought that organizations are generally satisfied with the overall quality of the software packages they purchase. Respondents generally agreed that software packages are reasonably reliable. As one respondent said:

I think companies are generally satisfied with the quality of applications; large outages are few and far between in comparison to the [19]90's. That being said, a smart customer will put reliability at the top of the list simply to pressure vendors to do more extensive QA at vendor expense vs. making the customer the beta site.

One respondent disagreed, however, observing that 'there is a perceived dissatisfaction with the overall quality of packaged software'.

Cost is the third most important attribute that enters managers' evaluations. The respondents in our study considered not only the upfront cost but also ongoing costs of keeping the

system running. One conjoint respondent emphasized that 'low total cost of ownership including software and hardware annual maintenance' entered his evaluation process. Another emphasized maintenance fees and support costs as part of the cost assessments. Another respondent observed that 'the cost of future customizations is always important'. Larger organizations with strong internal MIS staff seem to consider the upfront costs more than the total costs. One respondent from a large organization noted: 'I rarely need support for my products due to my size company and my abilities to handle issues. So, I look for vendors who are willing to reduce software support fees to be more in line with my demand of their support'.

It is noteworthy that functionality and reliability take precedence over cost. This suggests that managers explicitly consider cost, but that it is *not* the most important consideration in their decision-making. One respondent summarized her thoughts on the evaluation process in line with our empirical results, 'Capabilities [functionality] and reliability are the two top [factors] followed by cost'.

The fourth most important factor was ease of customization. There is anecdotal evidence that many packaged software implementations run into trouble because of difficulties that arise in customizing the software to the needs of the organization. Thus, we were surprised that ease of customization was not ranked higher. In our follow-up survey, we asked: 'Why do you think that MIS managers place a relatively low emphasis on ease-of-customization?' One respondent made the following observation:

This is basically the psychology of the manager. Most managers want to get moving and their target is to get the package selected and implemented. Ease of customization is something that comes up later in the life cycle.

Another respondent offered a different explanation:

The assumption is that if the right features are in the product up front, the need to customize is lower, and therefore ease-of-customization is not as much of an issue. This approach assumes a very thorough selection process; otherwise, the customer is left with a product that lacks critical features and will need extensive customization.

The fifth factor that was found to be significant was ease of use, but this was deemed to be the least important factor. In our follow-up survey, respondents concurred that the other factors were more important in evaluating enterprise packages.

Interestingly, two of the seven factors – ease of implementation and vendor reputation – were *not* found to be statistically significant. In the follow-up survey, we asked the MIS practitioners why they thought that these two factors did not appear to influence managers' purchase decisions. We think it is particularly interesting that ease of implementation was not found to be significant in spite of the fact that most of the obstacles that firms encounter with ERP software relate to implementation (Holland & Light, 1999; Ross, 1999; Ragowsky & Somers, 2002). With respect to ease of implementation, one respondent offered the following explanation:

I think this is because a significant number of enterprise software installations are implemented by third parties. This cost is not capitalized, and comes from an operations budget

not an IT capital budget. So, I think for some of the people surveyed, this is perceived as not coming from their budgetary pie.

It is entirely possible that ease of implementation may be an important factor, but one which is easily overlooked in the software acquisition process.

Another respondent noted that:

There is a strong belief that all of these packages are difficult to deal with . . . It's probably a nonfactor because the perceived differences in implementation effort between the competitors is small – they are all thought to be very difficult.

It is also possible that managers discount the costs associated with implementation, paying more attention to the immediate purchase price of the software. Thus, even if implementation costs are large for a given package, they may not have the same degree of influence as other factors at the time of purchase. Another interpretation for this pattern of results is that implementation attributes are less tangible and thus more difficult to assess objectively or to compare across prospective packages relative to other package attributes. Therefore, managers making the package acquisition decision may not ascribe as much weight to implementation issues as compared with other more immediate and observable attributes of the software.

We were also intrigued by the lack of significance associated with vendor reputation, because the literature clearly emphasizes this factor. Respondents to our follow-up survey offered several possible explanations, such as the following:

Anyone who has been in the IT industry for more than 10 years has seen the rise and fall of multiple reputations. Oracle almost went out of business in 1991, CA was once a pariah, and Sybase was considered a technical leader. Now, all three of those reputations have been inverted. So, I think a pragmatic manager ignores reputation; I would argue that this has become more commonplace as software is moving into more of a commodity phase.

Another respondent suggested that all of the vendors at this point are reputable and that it is the reputation of the implementer rather than the enterprise package vendor that matters most:

The market generally accepts that many of these packages are failure mature and the difficulty comes in tailoring it to their specific business rules and needs. Also, the list of vendors is relatively [short and the perception is that the industry 'shake-out' has already occurred (i.e. all of the fly by-night players are gone). For these reasons, I think the client generally is more concerned with the reputation of the implementer than that of the vendor.

5.1 Limitations

Before discussing the implications of these findings, we point out five limitations of the study. First, the enterprise system scenarios presented to the subjects were hypothetical. However, we would argue that the software buyer often faces situations in which s/he has to make a decision between two or more software packages that have a different combination of

strengths and weaknesses relative to the factors investigated here. So, in this sense, the decision situation we are modelling is quite realistic. While in reality, the buyer is presented with much more information to have to sort through than what we provided them with, we would argue that many companies eventually end up distilling all that information into a scorecard which is used to compare different packages across a set of key factors such as the ones investigated in this study. Further, we used prior literature to guide the choice of the attributes included in this study.

Second, a single-item measure was used to assess likeliness of recommending purchase. Although the use of single-item measures is common in conjoint studies, such measures cannot be assessed for reliability. Third, we measured intentions to recommend purchase – not actual purchase behaviour. Additional research is needed to test the extent to which intentions translate into actual behaviour. However, a vast body of work has shown that intentions and behaviours are tightly correlated (e.g. Mathieson, 1991; Taylor & Todd, 1995).

Finally, caution should be observed in generalizing our findings beyond the sample of MIS managers, as other organizational stakeholders may place different weights on the factors that were included in the study. Further work is needed to determine if business managers or end-users would have different evaluation criteria.

6 PRACTICAL IMPLICATIONS

The results obtained in this study have important implications for software development practice as well as for managers who are charged with making acquisition decisions for packaged enterprise software. The results not only provide a useful basis for buyers to assign weights to software package evaluation criteria, but also provide insights that should be useful to packaged software firms as they seek to produce enterprise-level software that will succeed commercially. The high degree of importance ascribed to reliability represents an interesting and novel finding. Given that software reliability is an important factor in shaping purchasing decisions, it is somewhat surprising that buyers are not exerting more pressure on software vendors to improve the reliability of their products. After all, a recent study conducted by the National Institute of Standards and Technology suggested that software bugs cost the US economy an estimated \$59.5 billion each year and that users incur 64% of these costs, or roughly \$38 billion/ year (Thibodeau, 2002). The results of our follow-up survey suggest that MIS practitioners are generally satisfied with the reliability of packaged software, but at the same time feel that vendors could do a better job in this area and do not appreciate being a beta test site.

6.1 Implications for package assessments by adopting organizations

In some cases, managers spend considerable time constructing an exhaustive list of selection criteria, but put relatively little time considering what weights to assign to the various criteria (Keil, 1998). When this occurs, important considerations like functionality tend to be assigned

less weight than they deserve. In effect, the importance of a particular factor can inadvertently be diluted in the evaluation process. Our results suggest that the five attributes that have statistically significant effects should be considered along with their weights in constructing a 'scorecard' for evaluating packaged software.

Table 3 provides an empirically derived scorecard for assigning weights to the criteria that should be considered in selecting packaged software. Project managers can use Table 3 to compare different packages and choose the one that offers the best weighted score for its bundle of attributes. The criteria and weights (derived from the conjoint analysis) are shown in Table 3 and should be used as a starting point, recognizing that each organization may want to customize both the criteria and weights used to evaluate enterprise packages.

The results of this study suggest that managers need to convey to software vendors the importance that they ascribe to software reliability. In general, this means doing a better job of raising the bar for software vendors with respect to what is considered to be an acceptable level of software reliability, and refusing to purchase software that does not measure up to that threshold.

6.2 Implications for software development organizations

The relative weights that we obtained should also be useful to organizations developing packaged software because they provide data from actual practitioners on the extent to which different attributes should be emphasized in the software development process. Here, it is particularly noteworthy that functionality and reliability are perceived to be more important than cost, ease of use, or ease of customization. It is also interesting to note that prospective buyers place functionality and reliability as having near equal importance. While software firms have traditionally been attuned to the importance of functionality, they have continued to offer software products that contain numerous bugs and which would be considered unreliable relative to other types of products. As one of the two most important factors in shaping managers' likelihood of recommending purchase, reliability can no longer be ignored by software developers. Given the high prevalence of software bugs and the costs that they pose for companies using packaged software (Mann, 2002), it would seem that managers who purchase software are placing more importance on software reliability than the software vendors. Perhaps this is because software users bear nearly two-thirds of the costs associated with software bugs, according to a recent study, while software vendors bear one-third of the cost (Thibodeau, 2002). While one would think that market forces would already have given development organizations the message regarding the factors that matter most to enterprise system purchasers, it is possible that the size and market power of some software companies have allowed them to avoid dealing with the issue of reliability.

Finally, our results suggest that managers do not place much emphasis on ease of implementation, in spite of the fact that this is where many ERP package implementation efforts fail. One possible explanation for this is that enterprise-level software packages are perceived as being uniformly difficult to implement. However, anecdotal evidence suggests that some packages are in fact easier to implement. Thus, additional research is needed to determine whether

managers are being short-sighted in ascribing such little importance to ease of implementation. Further research is also needed to determine the *process* that managers actually use to evaluate software packages and the extent to which they incorporate the criteria highlighted by this study in their decision-making process.

7 CONCLUSIONS

This study was motivated by the recognition that many organizations buy off-the-shelf systems instead of developing them from scratch. Surprisingly, prior research has made little attempt to examine what criteria MIS managers use in assessing such software. As a result, packaged software developers have sought to balance the trade-offs that exist between cost, quality and functionality, with little to guide them concerning what the customer actually values. This study breaks new ground by being the first conjoint study to examine empirically the relative importance that managers ascribe to various factors that are believed to be important in evaluating packaged software. The results provide insight into what MIS managers value most in acquiring such systems.

The results of this study have important implications for both organizations that use enterprise-level packaged software and the firms that develop such software. The results show that functionality, reliability, cost, ease of customization and ease of use – in that order – are the key criteria on which such software is judged. An especially noteworthy finding is the importance ascribed to reliability, which should serve as a wake-up call to the software industry concerning the need to improve software quality.

REFERENCES

- Appel, A., Dhadwaland, A. & Pietraszek, W. (2003) More bang for the IT buck. *Mckinsey Quarterly*, **2**, 130–141.
- ARC Advisory Group, Inc. (2003) ERP market opportunities change while remaining strong overall at \$8.9 billion. [WWW document] URL <http://www.arcweb.com/Community/arcnews/arcnews.asp?ID=328>
- Bernroider, E. & Koch, S. (2001) ERP selection process in midsize and large organizations. *Business Process Management*, **7**, 251–257.
- Brown, R.M. & Stephenson, K. (1981) The evaluation of purchased computer software. *Mid-South Business Journal*, **July**, 8–11.
- Bryce, M. & Bryce, T. (1987) Make or buy software? *Journal of Systems Management*, **38**, 6–11.
- Chau, P.Y.K. (1995) Factors used in the selection of packaged software in small businesses: views of owners and managers. *Information and Management*, **29**, 71–78.
- Goldenberg, B. (1991) Analyze key factors when choosing software. *Marketing News*, **25**, 23.
- Grudin, J. (1991) Interactive systems: bridging the gaps between developers and users. *IEEE Computer*, **24**, 59–69.
- Henfridsson, O. (2000) Ambiguity in IT adaptation: making sense of first class in a social work setting. *Information Systems Journal*, **10**, 87–104.
- Holland, C. & Light, B. (1999) A critical success factors model for ERP implementation. *IEEE Software*, **May/June**, 30–35.
- IT-Cortex (2003) Statistics over IT projects failure rate. [WWW document] URL http://www.it-cortex.com/Stat_Failure_Rate.htm

- Johannsen, C. (1980) Software selection criteria outlined. *Computerworld*, **February 4**, 33.
- Keil, M. (1998) Timberjack Parts: packaged software selection project. Harvard Business School, Video #9-399-510.
- Keil, M. & Carmel, E. (1995) Customer-developer links in software development. *Communications of the ACM*, **37**, 33–44.
- Keil, M. & Tiwana, A. (2005) The drivers of cots application value. *IEEE Software*, **22**, 64–69.
- Keller, E. (1994) It's easy to pick applications . . . Not! *Manufacturing Systems*, **January**, 50.
- Kintisch, R.S. (1987) Pick your package. *Management World*, **16**, 6–8.
- Kitchenham, B., Pfleeger, S., Pickard, L., Jones, P., Hoaglin, D., Emamand, D. *et al.* (2002) Preliminary guidelines for empirical research in software engineering. *IEEE Transactions on Software Engineering*, **28**, 721–734.
- Louviere, J. (1988) *Analyzing Decision Making: Metric Conjoint Analysis*. Sage Publications, Beverly Hills, CA, USA.
- Mann, C. (2002) Why is software so bad. *MIT Technology Review*, **July/August**, 33–38.
- Mathieson, K. (1991) Predicting user intentions: comparing the technology acceptance model with the theory of planned behavior. *Information Systems Research*, **2**, 173–191.
- Mingers, J. (2001) Combining IS research methods: towards a pluralist methodology. *Information Systems Research*, **12**, 240–259.
- Montazemi, A.R., Cameron, D.A. & Gupta, K.M. (1996) An empirical study of factors affecting software package selection. *Journal of Management Information Systems*, **13**, 89–105.
- Murphy, K.E. & Simon, S.J. (2002) Intangible benefits valuation in ERP projects. *Information Systems Journal*, **12**, 301–320.
- Pivnicny, V.C. & Carmody, J.G. (1989) Criteria help hospitals evaluate vendor proposals. *Healthcare Financial Management*, **43**, 38–43.
- Priem, R. (1992) An application of metric conjoint analysis for the evaluation of top managers' individual strategic decision making processes: a research note. *Strategic Management Journal*, **13**, 143–151.
- Ragowsky, A. & Somers, T. (2002) Enterprise resource planning. *Journal of Management Information Systems*, **19**, 11–16.
- Romanow, D., Keil, M. & McFarlan, F.W. (1998) Timberjack Parts: packaged software selection project. Harvard Business School, Case #9-398-085.
- Ross, J. (1999) Surprising facts about implementing ERP. *IEEE IT Professional*, **July/August**, 65–68.
- Sawyer, S. (2001) Information systems development: a market-oriented perspective. *Communications of the ACM*, **44**, 97–102.
- Shang, S. & Seddon, P.B. (2002) Assessing and managing the benefits of enterprise systems: the business manager's perspective. *Information Systems Journal*, **12**, 271–299.
- Sherer, S.A. (1993) Purchasing software systems: managing the risks. *Information and Management*, **24**, 257–266.
- Software & Information Industry Association (2003) Packaged software industry revenue and growth. [WWW document] URL <http://www.siiia.net/divisions/research/growth.asp>
- Stensrud, E. & Myrtveit, I. (2003) Identifying high performance ERP projects. *IEEE Transactions on Software Engineering*, **29**, 398–416.
- Taylor, S. & Todd, P.A. (1995) Understanding information technology usage: a test of competing models. *Information Systems Research*, **6**, 144–176.
- Thibodeau, P. (2002) Study: buggy software costs users, vendors nearly \$60B annually. *Computerworld*, **June 25**, [WWW document] URL <http://www.computerworld.com>

Biographies

Mark Keil is the Board of Advisors Professor of Computer Information Systems (CIS) in the J. Mack Robinson College of Business at Georgia State University. His research focuses on software project management and has appeared in *MIS Quarterly*, *Sloan Management Review*, *Communications of the ACM*, *Journal of Management Information Systems*, *IEEE Transactions on Engineering Management*, *Information Systems Journal*, *Decision Support Systems*, *Information & Management* and many other journals. In 2005, he received the Georgia State University Alumni Distinguished Professor Award for outstanding achievements in scholarship, teaching and service. He currently serves on the editorial boards of *IEEE Transactions on Engineering Management*, *Decision Sciences* and the *Journal of Management Information Systems*. He has also served as an Associate Editor for *MIS Quarterly* and as Co-editor of *The DATA BASE for Advances in Information Systems*. He earned his bachelor's degree from Princeton University, his master's degree from MIT's Sloan School of

Management, and his doctorate in management information systems from the Harvard Business School.

Amrit Tiwana is an Assistant Professor in Iowa State University's College of Business in Ames, Iowa, USA. His PhD is from Georgia State University's Robinson College of Business. He was on the faculty of Emory University in Atlanta when this research was completed. His research focuses on knowledge management in software develop-

ment. His work has appeared in journals including *California Management Review*, *Journal of Management Information Systems*, *IEEE Transactions on Engineering Management*, *IEEE Software*, *IEEE Internet Computing*, *Communications of the ACM*, *Decision Support Systems*, *Information Systems Journal*, *Journal of Knowledge Management*.

APPENDIX: SURVEY INSTRUMENT

The purpose of this study is to understand how MIS managers manage cost, quality and functionality trade-offs in evaluating packaged enterprise software applications. Examples of such packages include customer relationship management (CRM) and enterprise resource planning (ERP) applications.

INSTRUCTIONS: The tables below describe eight enterprise software packages *relative to comparable packages*. Based on this information and your own expertise, please answer the two questions next to each table. Here,

- **FUNCTIONALITY** refers to extent to which the package contains all the features and functions specified in your RFP.
- **RELIABILITY** refers to overall quality and stability of the software.
- **COST** refers to the total software acquisition + implementation cost.
- **EASE OF USE** refers to the intuitiveness of the user interface.
- **VENDOR REPUTATION** refers to the reputation that the vendor has established for unconditionally standing behind its products.
- **EASE OF IMPLEMENTATION** refers to the set-up time and effort that will be required to implement this package in your organization.
- **EASE OF CUSTOMIZATION** refers to the extent to which this package can be easily modified to meet your organization's unique needs.

Sample enterprise software package profile

(A total of eight such software package profiles developed using the CONJOINT algorithm implemented in SPSS were provided to each organization; all profiles resembled the sample profile shown below but had different attribute-level combinations.)

Enterprise package one of eight												
Functionality	Low	What is the likelihood that you would recommend that your organization acquire this package?										
Reliability	Low											
Cost	Low											
Ease of Use	High											
Vendor reputation	High											
Ease of implementation	High											
Ease of customization	High											
												Very low

Conjoint profiles used in the study

Conjoint project profile	Functionality	Reliability	Cost	Ease of use	Vendor reputation	Ease of implementation	Ease of customization
1	Low	Low	Low	High	High	High	High
2	Low	High	High	Low	High	High	Low
3	High	High	Low	High	Low	High	Low
4	Low	Low	Low	Low	Low	Low	Low
5	High	Low	High	High	High	Low	Low
6	Low	High	High	High	Low	Low	High
7	High	High	Low	Low	High	Low	High
8	High	Low	High	Low	Low	High	High

MARKED PROOF

Please correct and return this set

Please use the proof correction marks shown below for all alterations and corrections. If you wish to return your proof by fax you should ensure that all amendments are written clearly in dark ink and are made well within the page margins.

<i>Instruction to printer</i>	<i>Textual mark</i>	<i>Marginal mark</i>
Leave unchanged	... under matter to remain	Stet
Insert in text the matter indicated in the margin	⤴	New matter followed by ⤴
Delete	⤵ through matter to be deleted	⤵
Delete and close up	⤵ through matter to be deleted	⤵
Substitute character or substitute part of one or more word(s)	/ through letter or ⤵ through word	New letter or new word
Change to italics	— under matter to be changed	⤵
Change to capitals	≡ under matter to be changed	≡
Change to small capitals	= under matter to be changed	=
Change to bold type	⤵ under matter to be changed	⤵
Change to bold italic	≡ under matter to be changed	≡
Change to lower case	Encircle matter to be changed	⊖
Change italic to upright type	(As above)	⤵
Insert 'superior' character	/ through character or ⤴ where required	⤴ under character e.g. ⤴
Insert 'inferior' character	(As above)	⤵ over character e.g. ⤵
Insert full stop	(As above)	⦿
Insert comma	(As above)	,
Insert single quotation marks	(As above)	⤴ and/or ⤵
Insert double quotation marks	(As above)	⤴ and/or ⤵
Insert hyphen	(As above)	Ⓜ
Start new paragraph	⤴	⤴
No new paragraph	⤵	⤵
Transpose	⤴	⤴
Close up	linking ⦿ letters	⦿
Insert space between letters	⤴ between letters affected	#
Insert space between words	⤴ between words affected	#
Reduce space between letters	⤴ between letters affected	⤴
Reduce space between words	⤴ between words affected	⤴