The Effects of Customizability and Reusability on Perceived Process and Competitive Performance of Software Firms

Author(s): Sarma R. Nidumolu and Gary W. Knotts
Source: *MIS Quarterly*, Vol. 22, No. 2 (Jun., 1998), pp. 105-137
Published by: Management Information Systems Research Center, University of Minnesota
Stable URL: https://www.jstor.org/stable/249392
Accessed: 03-09-2018 16:54 UTC

The Effects of Customizability and Reusability on Perceived Process and Competitive Performance of Software Firms[1]

By: Sarma R. Nidumolu
Department of Management Information Systems
College of Business and Public Administration
McClelland Hall, University of Arizona
Tucson, AZ 85721
U.S.A.
rnidumolu@bpa.arizona.edu

Gary W. Knotts
Department of Management Information Systems
College of Business and Public Administration
McClelland Hall, University of Arizona
Tucson, AZ 85721
U.S.A.
gknotts@bpa.arizona.edu

Abstract

This study addresses the broad research issue of how software firms can manage their software development efforts in order to compete effectively under intensified competition. Based on recent research in manufacturing strategy and software process engineering, a research model and six hypotheses were derived. Reusability and customizability were expected to positively affect process flexibility and predictability. In turn, these perceived process performance dimensions were expected to positively influence perceived competitive performance, assessed in terms of market responsiveness and product cost efficiency.

Using a survey design, responses were obtained from a random sample of 100 software firms. Two kinds of respondents were used: the senior manager in charge of software development (58% response rate) and the marketing manager (36% response rate). The model and hypotheses were assessed using EQS, a structural equations modeling package that can be used for path analysis. The results from both the marketing manager and development manager responses suggested that process flexibility was an important determinant of perceived competitive performance. However, process predictability was an important determinant of perceived competitive performance in the development manager, but not the marketing manager, responses. Finally, while customizability had a significant positive effect on the perceived process performance dimensions, reusability did not. The research model is a potentially useful contribution to an important new area of MIS research concerning the performance of software firms, which draws from manufacturing strategy and software process engineering.

Keywords: Software process management, customizability, reusability, process predictability, process flexibility, perceived competitive performance

ISRL Categories: AF0405, EE03, EL03, FC15

Motivation

How can software firms manage their software development efforts in order to compete effec-

[1]Lynda Applegate was the accepting senior editor for this paper.

tively in today's dynamic and uncertain environment? It is important to consider this broad research issue for several reasons. First, the U.S. software industry is significant in its own right, employing almost two million workers and with sales of software estimated at almost $103 billion in 1996 (Carricaburu 1997). As a symbol of the Information Age, perhaps even its first true business activity, the software industry holds a special fascination with the public.

Second, the software industry has an important influence on the economy:

> As the twentieth century draws to a close, software is a driving force in the operations of businesses, governments, and the military. Most businesses in the industrialized world could not compete, and many could not even survive, without computers and software (Jones 1994, p. 17).

Recent research suggests that information technology investments have resulted in substantial productivity gains for business firms. Consequently, improvements in the software industry's development efforts are bound to impact the economic efficiency of these businesses and therefore of the economy as a whole. Finally, many software firms confront the need to respond quickly to market changes through reduced time-to-market of software. Such pressures, and opportunities for quick-responding firms, are similar to those facing firms in many other industries where competition, market demands, and technological changes have intensified in recent years. Consequently, lessons learned from the software industry about how to produce software to increase market responsiveness can also be useful for other industries facing similar market pressures.

In practice, a variety of means can be used to improve the performance of a software firm, including new technologies such as object-oriented computing, methodologies such as the spiral model for managing software development projects, and approaches that consider the underlying process used across projects. While each of these means is important, this study focuses particularly on the latter because of the considerable attention that process-based approaches have received recently in industry and academia. Important examples of such approaches include the capability maturity model (CMM) of the Software Engineering Institute (Humphrey 1989) and the software factory approach used mostly by Japanese firms (Cusumano 1991). The software development process refers to the set of tools, methods and practices used to produce software products (Humphrey 1989, p. 3). Based on prior research in software process engineering, two key objectives of process-based approaches were identified: reusability and customizability. The former refers to the extent to which the outputs of a software development project can be reused for other projects. The latter refers to the extent to which a software development approach can be tailored to the needs of individual projects. The purpose of this study is to identify how these two objectives affect the performance of the software firm, through their effect on process performance.

Since firm performance is a broad concept, this study particularly focuses on perceived competitive performance, which describes the firm's perceived performance relative to its competition. By drawing from manufacturing strategy research, two dimensions of perceived competitive performance are considered: customer responsiveness and product cost efficiency. The former refers to the organization's relative ability to respond to market changes whereas the latter describes the relative efficiency with which the organization produces its products. By drawing additionally from software process engineering research, two dimensions of perceived process performance are identified: process predictability and flexibility. The former describes the organization's ability in using the software development process to accurately estimate the resources needed for developing software products, whereas the latter describes the speed with which the software development process can be used to respond effectively to changes in the organization's environment. The specific research question addressed in this study is: *What is the effect of reusability and customizability on software process flexi-*

bility and predictability, and on the software firm's perceived competitive performance?

Research Model

In addressing the research question posed above, this study draws from two distinct, but related, research areas. Manufacturing strategy research has long focused on the competitive performance implications of the processes used to manufacture products, whereas software process engineering researchers (e.g., Cusumano 1991, pp. 440–444; Humphrey 1989, p. 3) have drawn significant comparisons between developing software and manufacturing products. The research model (Figure 1) presented in this study conceptually synthesizes relevant research in these two areas and suggests that a software firm's ability to meet the objectives of reusability and customizability significantly influences perceived software process performance, which in turn affects the perceived competitive performance of the firm. The constructs used in the research model and their inter-relationships are described below.

Perceived competitive performance

In this study, a software firm's perceived performance relative to its competitors was the focus. Such comparisons are feasible because a high degree of market segmentation has been "an enduring feature of industry structure" in the U.S. software industry, with Microsoft and a few other companies being exceptions (Mowery 1996, p. 11). Two important aspects of perceived competitive performance derived from manufacturing strategy research are product cost efficiency and market responsiveness. The former describes the efficiency with which the organization produces its products (Nemetz and Fry 1988), whereas the latter describes how timely the organization is in responding to market changes (Boynton et al. 1993).

Since performance is complex and multidimensional, the choice of dimensions to describe performance will depend on the viewpoint adopted. In this study, the emphasis was on studying how a process-based approach to software development, analogous to manufacturing processes in product-based industries, influenced performance. Product cost efficiency and market responsiveness were therefore selected because they illustrate key advantages of mass production and job shops respectively and were traditionally considered to entail significant tradeoffs. However, they need to be simultaneously pursued if firms are to compete effectively in today's dynamic and uncertain environment (Nemetz and Fry 1988). For example, while rapid entry into a new market is important, pioneering firms which do not target the mass market subsequently are likely to fail (Golder and Tellis 1993). These constructs are defined in greater detail below.

Product cost efficiency. This concept has been an important characteristic of mass production firms (Nemetz and Fry 1988). A key advantage of new manufacturing approaches such as mass customization is that firms can continue to obtain the cost efficiencies that characterized mass producers (Boynton et al. 1993). In this study, product cost efficiency of a software firm is defined as its ability, relative to competitors, to produce software products efficiently from a cost perspective.

Market responsiveness. This concept is important in coping with demand uncertainties for a firm's products and is a critical performance goal of firms which pursue mass customization (Pine 1993, p. 189). In this study, market responsiveness of a software firm is defined as its speed, relative to competitors, in responding to changes in marketplace demands for its software products.

Dimensions of perceived process performance

The choice of dimensions for measuring perceived *process* performance depends significantly on the dimensions of firm performance

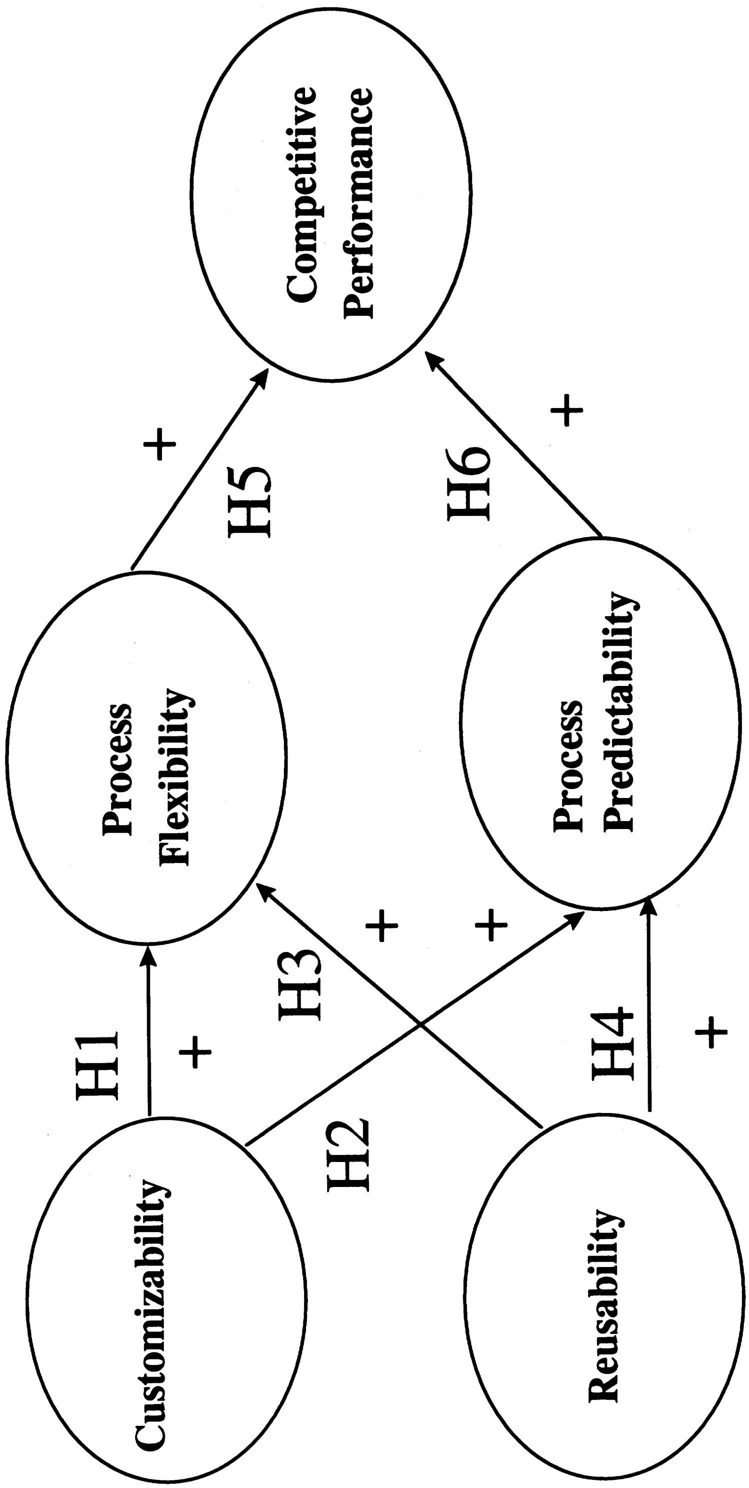

Figure 1. Research Model

that are emphasized. In product manufacturing, process flexibility and predictability have traditionally been associated with the tradeoffs perceived between craft/job-shops and mass production. Manufacturing process flexibility is the primary means by which a firm can rapidly respond to changes in markets and technologies and is especially relevant given the changing conditions confronting manufacturing organizations (Jordan and Graves 1995). At the same time, high levels of predictability characterized mass production firms that were under statistical process control. However, recent developments in manufacturing suggest that firms could have production processes that are both flexible and predictable.

Process flexibility. In manufacturing strategy research, flexibility can be studied at a number of levels and domains of application. Although a variety of flexibilities can be defined, these can broadly be classified into two types: speed of response to the environment and the scope or range of responses that can be generated (Parthasarthy and Sethi 1992). This study considers the former aspect, because it is becoming a critical measure of manufacturing process performance in general, and software product development in particular (Olsen 1995). In this study, software process flexibility is defined as the speed with which the organization's software development approach can respond effectively to changes in the organization's environment.

Process predictability. A stable manufacturing process has been the critical characteristic of mass production firms, which were designed to cope with as little change as possible (Boynton et al. 1993). In software engineering, software process predictability is critical to the survival of software firms, given the importance of delivering a product within budget and on time. For example, firms whose costs, schedules, and performance are unpredictable are considered at the lowest level of the CMM (Humphrey 1989, p. 5). In order to advance to the next maturity level, these firms need to be able to meet their cost, time, and performance estimates. In this study, software process predictability is therefore defined as the ability of the software development organization to accurately estimate the needed resources, time, and performance of its software projects (Dowson 1993).

Characteristics of software development approach

Software process engineering research has identified several features that characterize a process-oriented approach. These include systematic collection and analysis of process-related information, periodic update of standards, development and use of an integrated package of methods and controls applied across projects, systematic accumulation of software development experience, extensive training to standardize employee skills, and detailed software process modeling. The CMM additionally describes several characteristics that establish the level of software process maturity (Paulk et al. 1993). It is difficult in practice to explicitly consider a long list of process-oriented characteristics to clearly describe the extent to which a software development approach is process-oriented. In order to study the performance impacts of process-oriented approaches, this study therefore focuses instead on some important objectives of process orientation that link the characteristics with perceived process and competitive performance. The ensuing research model is expected to be parsimonious and explain a greater variance in performance than when a (necessarily incomplete) list of process-oriented features is considered.

In software development, both economies and diseconomies of scale are present, suggesting a non-linear relationship between software development effort and the size of the developed application (Banker et al. 1994; Hu 1997). In addition, economies of scope are also being recognized as a basis for improved software development process performance because significant benefits accrue from reapplying resources across more than one project (Cusumano 1991, pp. 8, 427). These resources include people, methods, tools, and intermediate outputs generated during software development. Manufacturing research in the last two decades also suggests that the tradeoffs traditionally perceived between the performance dimensions described can be mitigated through economies of scope, where resources used in one product can be reapplied to other products (Panzar and Willig 1981). This study chose two important objectives of process orientation—reusability and customizability—for further research because they illustrate such economies of scope and have also been the focus of much research in software process engineering, as described below.

Reusability. In general, reusability illustrates the economies of scope that results from recycling the outputs from a software development project to another project. In software process engineering, interest in reusability is currently very significant, comparable to that for structured techniques in the early 1980s. It has been perceived by some experts as the "only realistic, technically feasible solution" to important problems of software development (Mili et al. 1995, p. 529). Reusability has been the primary objective of software factories in Japan (Cusumano 1991, p. 32), and a variety of field studies suggests that it is an important means of improving software development performance in the U.S. as well (Banker and Kauffman 1991). Reuse need not be confined to code only; other outputs from every phase of the project can also be reused. In this study, reusability is defined as the extent of reuse of outputs generated during analysis, design, coding, testing, and other phases of development (Apte et al. 1990).

Customizability. This concept refers generally to the software development unit's ability to reapply its software development approach to a wide variety of purposes and circumstances, therefore resulting in economies of scope. Customizability is an essential objective for a software process because it enables the process to be dynamically adjusted to the unique needs of each project (Humphrey 1989, p. 247). Customized software processes are emphasized by the CMM, particularly at the two highest levels of process maturity (Paulk et al. 1993). In this study, customizability is defined as the ability of the software development approach to be tailored to the specific needs of individual projects.

Missing variables

In attempting to develop a parsimonious model that describes the effects of reusability and customizability on perceived process and competitive performance, it is likely that a number of missing variables may serve as potential confounds. For example, predictability and customizability could depend on the nature of the software products (e.g., size, complexity, type, and homogeneity) produced by the firm, the nature of the personnel resources (e.g., size and composition of the teams) that are employed, and the nature of the markets in which the firm competes. However, there are at least three considerations that make the model useful to evaluate even without these variables: (1) the primary intent of the model is not so much to explain perceived process or competitive performance, but to establish whether reusability and customizability—two key objectives of process-based approaches—are significant predictors of perceived performance; (2) by choosing variables that are linked by a common theoretical thread, i.e., economies of scope that mitigate traditionally perceived tradeoffs between mass production and craft/job-shops, a stronger causal chain between the variables may be established; and (3) the effect of missing variables can be reduced somewhat by careful choice of methodological controls, as illustrated in the research methodology section.

Research hypotheses

Traditionally, software developers achieved the flexibility of craft or job-shop production by developing software on a project-by-project basis. However, such an arrangement did not compare favorably with the reliability and productivity of mass production (Cusumano 1991, p. 441). Software process engineering research suggests that flexibility can also be achieved by a software process that is customizable. It acts as a template that provides explicit guidance on how to undertake individual software development projects (Curtis et al. 1992), but also helps strike a balance between total permissiveness and rigid enforcement (Dowson 1993). Customizability enables the software developer to dynamically identify the best process solution to the specific problem (Bandinelli et al. 1993). Many Japanese software developers have processes that are customizable to individual projects, which allows them to adapt to environmental pressures such as shortages of programmers or rapid growth in demand for custom applications (Cusumano 1991, p. 41). At the two highest levels of the CMM model, firms can tailor their standard development processes to apply to specific project requirements and improve their ability to react to the environment (Paulk et al. 1993). Overall, by leveraging previous knowledge, a customizable software development approach can therefore be used to respond rapidly to environmental changes. These arguments suggest:

H1: Increases in customizability are associated with increases in process flexibility.

Traditional mass production was based on *"efficiency through stability and control* . . . if this is achieved, all else will follow" (Pine 1993, p. 28, emphasis in the original). Such stability was achieved by using a standardized process to produce homogeneous products using large batch sizes with very little retooling (Nemetz and Fry 1988). The need to customize could compromise the reliability and control of the process. However, research in the last decade suggests that customizability and process predictability need not be antithetical, particularly in software development. For example, a development approach that can be customized to meet the needs of individual projects could also promote greater control and optimization because of an understanding of cause-effect relationships (Humphrey 1989, p. 449). Because knowledge in tailoring the process to one project can be easily transferred to other projects, exceptions in process outcomes are less likely and more noticeable. Overall, in contrast with early research, this study suggests:

H2: Increases in customizability are associated with increases in process predictability.

There is some evidence that reusability significantly reduces software development cycle time and hence increases the flexibility of the process. For example, the implementation of reusability at Celite Sales Corporation (described by Swanson et al. 1991, p. 575) led to significant improvements in process flexibility by enabling "dramatic reductions in development time." The reduced development times were primarily because of increases in software developer productivity as a result of reusing pre-existing components. As another example, programming productivity at the Toshiba Software Factory in Japan increased by 57% between 1979 and 1985, due primarily to a 270% increase in code reuse levels in the same period (Cusumano 1991, p. 240). Reusability also increases process flexibility by making it easier to modify the developed software, which can be even more beneficial than faster development times. Software development units that reuse components extensively can be likened to flexible factories that can rapidly assemble the required product (Cusumano 1991, p. 258). Overall, software engineering research on reusability suggests that:

H3: Increases in reusability are associated with increases in process flexibility.

Reusability can also lead to significantly higher levels of reliability, therefore resulting in a process with reduced variation in error rates. Because reuse of components provides greater opportunities for error discovery and correction, the number of unexpected problems decrease over time and the software development process becomes more stable. As has been suggested, "reusable parts are very predictable once programs pass their initial tests . . . if the process is followed, problems usually found in a system test can be avoided" (Swanson et al. 1991, p. 575). The more the software component is reused, the more its capabilities become known and the less the likelihood of errors (Reifer 1992). Japanese software factories achieved high levels of process predictability because of a systematic program of increasing reusability (Cusumano 1991, pp. 191, 436). These arguments suggest that:

H4: Increases in reusability are associated with increases in process predictability.

Traditionally, manufacturing flexibility was assumed to negatively impact cost efficiencies in producing products (Nemetz and Fry 1988). However, this efficiency-flexibility tradeoff has increasingly been questioned by both manufacturing research in the last decade (Kotha 1995) and recent MIS research (Boynton et al. 1993). Manufacturing flexibility, obtained through advances in manufacturing automation and organization, is a key means for increasing market responsiveness when future demand is uncertain and competition has intensified (Jordan and Graves 1995). At the same time, economies of scope through sharing of resources ensure that cost efficiencies need not be compromised (Kotha 1995). This suggests:

H5: Increases in process flexibility will be associated with increases in perceived competitive performance of the organization.

For a large number of industries, the competitive environment requires revolutionary product change, but also permits firms to improve their manufacturing processes on an incremental and continuous basis (Boynton and Victor 1991). Firms in such industries can be responsive to their customers without having to abandon the cost advantages of mass production, therefore succeeding through "dynamic stability" (Boynton et al. 1993). Software development processes are especially amenable to incremental refinement, as illustrated by the popularity of continuous process improvement approaches such as total quality management in software, the CMM, and software factories. A predictable process is the basis for making such long-term improvements as it enables firms to systematically and continuously improve process know-how and capabilities (Boynton and Victor 1991). These arguments suggest that:

H6: Increases in process predictability will be associated with increases in perceived competitive performance of the organization.

Research Methodology

Sampling strategy

The sampling frame for this study was the American Software Association (ASA) Division of the Information Technology Association of America (ITAA). The ITAA was selected because it is one of the most diversified in the country and represents information technology firms that range in size from a few people to thousands of employees. The ASA is one of five divisions within the ITAA and includes members who range from large software firms such as Microsoft, IBM, SAP America, and Oracle to small software firms. The ASA thus provides a wide spectrum of software developers from which to sample. However, response rates were expected to be problematic given the intense competition in the software industry and the sensitivity of the issues such as process flexibility and predictability. Since resources were limited, an intensive sampling strategy was selected, where repeated contacts with a smaller, but more manageable, sample was expected to be more cost efficient than fewer contacts with a larger sample.

The study questionnaire was mailed to a random sample of 100 firms selected from the approximately 350 software firms who were members of the ITAA's ASA Division in 1993-94. On an average, eight follow up contacts were made with each firm in the sample, resulting in 58 questionnaires usable for analysis (58% response rate). The key informant who received the survey was typically the VP (Development) or chief technology officer

(CTO). The median software firm in the sample (Table 1) was in business for the past 15 years, employed 35 full-time employees, grossed $3 million in sales annually, introduced about one new product each year, and had been using its current software development approach for the past six years. The sample included firms such as IBM, Trinzic Corporation, and Hyperion Software that produce software for Fortune 500 firms across a wide variety of industries. Also included were niche firms such as Reynolds and Reynolds Healthcare Systems, which develops patient accounting systems for doctor's offices and medical clinics, Pentamation Software, which develops software for K-12 schools and local governments, and First Resort Software, which develops software for property management companies.

Single-respondent samples, although common in MIS research, have potential for bias because of the respondent's lack of knowledge, leading to guessing or random selection of a score. In this study, since software development is the core technology of a software firm, the VP (Development)/CTO was likely to be knowledgeable about the firm's competitive positioning vis a vis process performance. However, there could be biased attribution of perceived competitive performance outcomes because of the respondent's vested interest in the causal factors. Consequently, second-respondent evaluations of perceived competitive performance were also collected in this study. The respondents from the first sample were asked to identify their counterparts in marketing who could answer the performance-related questions. On an average, about four follow-up contacts had to be made with each firm. Altogether, 36 usable second-respondent questionnaires (62% response rate) were returned from the 58 firms in the first sample.

Questionnaire development

Because many constructs had not been studied empirically in previous MIS research, measurement scales had to be developed by identifying individual concepts emphasized by researchers. These concepts were then used to create individual items and the measurement scales were subsequently subject to a variety of analyses for assessing their reliability and validity. The scales were first assessed for content validity, which describes the extent to which the items contained in an instrument were sampled from the universe of all possible items for a particular construct. In this study, expert review was used to verify content validity during a pretest, where the measurement items for the constructs were assessed by five academics specializing in information systems and 10 software development professionals in

Table 1. Sample Characteristics

Aspect	Mean	S.D.	Median	Min.	Max.
Number of years in business	20.2	18.7	15	2	100
Number of full-time employees	608.9	1,881.3	35	2	12,000
Annual sales volume ($ million)	2.6	1.6	3	0	11,000
Number of software projects undertaken per year	37.2	75.4	10	0	400
Percentage of software development outsourced	8.3	15.2	0	0	80
Number of new products introduced per year	8.3	23.8	1	0	122
Number of years software development process in use	7.6	6.1	6	1	25

N = 58 firms.

industry. All reviewers completed the questionnaire and also responded to the following questions

(1) Were there items that should have been included on the questionnaire, but were not? (2) Were there items on the questionnaire that should be deleted because they were irrelevant? (3) Were there items on the questionnaire that should be reworded to reduce ambiguity? Upon completing the questionnaire, the reviewers were interviewed and asked to describe their interpretation of the questions, in order to compare them with the original intent. The final questionnaire, modified as a result of the pretest, was then used for the survey. The study's measurement scales are given in Appendix A.

Measures

Perceived competitive performance. This construct was given by the respondent's assessment of the software firm's position relative to its competitors (1 - at the very bottom, 3 - in the middle, 5 - at the very top) on the following two dimensions identified earlier (Nemetz and Fry 1988; Pine 1993, p. 189):

1. *Product cost efficiency:* (a) ability to produce software at low cost for current product lines, (b) ability to charge competitive prices for software for current product lines, (c) the efficiency of software production for current product lines, and (d) the productivity of software developers for current product lines.

2. *Market responsiveness:* (a) the speed of response to new customer needs, (b) ability to tailor software products to individual customer needs, (c) the speed at which new software markets can be entered, and (d) the rate of introduction of new software products/services.

The items in each dimension were identified in field interviews prior to and during the pretest. In general, performance has traditionally been particularly problematic to measure, both in organization studies and in MIS research. In this study, a major problem was the lack of quantitative performance measures such as return on investment (ROI), return on net assets (RONA), and earnings per share (EPS) for 42 of the 58 firms in the sample. These firms were privately held and therefore not obliged to file their performance data with the SEC. The three-year average scores for ROI, RONA, and EPS for 1991-94 were obtained for the other 16 (publicly held) firms.

To cross validate the two-dimensional scale above, the study also used a unidimensional subjective scale originally developed by Dess and Robinson (1984) and recommended in situations where objective financial performance data are hard to get, such as for private firms. The individual items in this unidimensional scale were evaluated by the respondent in relation to the software firm's competitors (see Appendix A). Moreover, to reduce single-respondent bias, the two-dimensional and unidimensional subjective scales were administered to both the VP (Development)/CTO and the VP (Marketing). In summary, a variety of scales were used to obtain a balanced assessment of perceived competitive performance of the software firm.

Process flexibility. Respondents were asked to indicate the speed, when compared to competitors in the same situation (1 - slower than all, 2 - slower than most, 3 - in the middle, 4 - faster than most, 5 - faster than all), with which their current software development approach could be used to respond effectively to changes in the following five business environment dimensions (Bantel 1993):

1. *Labor supply:* (a) quantity of developers in the labor market decreased, and (b) quality of developers in the labor market decreased.

2. *Customer needs:* (a) customer's software product requirements changed, and (b) a new market for the software products opened.

3. *Competition:* (a) new competitors entered the software market, and (b) changes in the basis of competition in the industry occurred.

4. *Regulation:* (a) new laws regulating the business were enacted, or (b) laws regulating the business were repealed.

5. *Technology:* (a) new software technologies became available, and (b) new hardware technologies became available.

Process predictability. This construct was measured by the extent to which each of the following were predictable (1 - very unpredictable, 5 - very predictable) at the start of a software development project in their organization (Henderson and Lee 1993; Keller 1994):

(1) actual date of completion of project, (2) actual developer-months required for the project, (3) actual budget that would be consumed by the project, (4) actual resources that would be required to complete the project, (5) the actual quality of the finished software, and (6) the actual functionality of the finished software.

Reusability. There is considerable controversy and little agreement in the literature over how to measure reusability. Some of the issues include what kinds of outputs should be considered as valid components for reuse, what kinds of measurement scales could be employed, whether reuse for versions of the same product should be allowed, and whether levels or quality of reuse should be used. In this study, the approach to measuring reusability for the first two issues was kept deliberately broad, i.e., valid reusable components included not just code, but also specifications, data, and documentation. Moreover, both interval and ratio scales for measuring reuse were used to permit cross-validation of results. However, much reuse occurs between versions of the same product, which could misleadingly signify economies of scope for producing a wide range of products. Consequently, this study focused only on interproduct, rather than intraproduct, reuse as a measure of the recycling of outputs from one software development project to another. Moreover, the focus was on level, rather than quality, of reuse because it was less likely to be confounded with performance.

Specifically, level of reuse was operationalized in terms of the respondent's estimate of the percentage (a ratio scale) of each of the following outputs of software development which consisted of reused components (Cusumano 1991, p. 11):

(1) requirements specifications, (2) design specifications, (3) software code, (4) test data, and (5) documentation. In addition, the study also used a 5-point Likert scale (interval scaled) which elicited the respondent's perceived extent of reuse (1 - not at all, 3 - to some extent, 5 - to a great extent) for each of the five items identified above. In answering the questions, the respondent was asked specifically to exclude reuse between versions of the same software product, for reasons described above.

Customizability. This construct was given by the extent to which the respondents agreed (1 - strongly disagree, 3 - neither agree nor disagree, 5 - strongly agree) with the following statements concerning their development approach:

(1) it could be used effectively for a variety of projects, (2) it could be customized by individual project teams to suit their context, (3) it permitted a range of responses to different types of requirements, (4) it permitted a range of responses to different types of technologies, (5) it could be adapted to the quality goals of individual projects, (6) it could be adapted to the productivity goals of individual projects, and (7) it could be adapted to suit the resource levels available to a project.

Analytical techniques

The research model and hypotheses were tested using path analysis, where each variable in the model was measured by a single indicator. The indicator was constructed as follows:

(1) where the variable was a unidimensional construct (e.g., customizability and process predictability), the indicator was given by the mean of the items that measured the construct, after testing for construct validity and reliability and (2) where the variable was multidimensional (e.g., process flexibility and perceived competitive performance), principal component analysis was used to reduce the dimensions into a single value given by the first principal component. In this study, path analysis was undertaken using structural equation modeling (SEM) techniques. SEM-based path analysis assumes that the variances and covariances of the independent variables are also parameters to be estimated, in contrast to traditional path analysis using multiple regression which assumes them to be known numbers. Consequently, SEM-based path analysis provides better estimates of the path coefficients than traditional techniques.

The specific SEM-based technique used in this study was EQS (Bentler 1989), which can be used for both path analytic and latent variable modeling. EQS is a viable alternative to LISREL and even provides some advantages such as a simpler variable classification scheme, which in turn results in a considerably simplified set of symbols for representing variables. In addition, EQS directly reports the results of sophisticated model specification techniques such as the Lagrange Multiplier (LM) and Wald tests that need to be computed in many other packages. In EQS, a variety of statistics are used to assess the overall empirical support for the research model. First, if the model's conformance with the observed data is to be considered acceptable, the probability value for the model's chi-square statistic should exceed a standard cutoff of 0.05 (Bentler 1989, p. 37). In addition, three fit indices are typically used to identify goodness of fit: (1) the comparative fit index (CFI), (2) the normed fit index (NFI) such as the Bentler-Bonett NFI, and (3) a non-normed fit index (NNFI). Values greater than 0.90 are desirable for each fit index. Next, the average absolute standardized residual (AASR) indicates the proportion of the variance not explained by the model and should have small values. Finally, the theorized model should be compared with a more constrained model, where previously free effects are now fixed, and a less constrained model, where previously fixed effects are now free to be estimated. The former comparison is undertaken in EQS by the Wald test, while the latter is performed by the LM test. Support for the model is enhanced if both the Wald and LM tests do not suggest any modifications that would significantly improve fit.

Control for missing variables

In an earlier section, several missing variables were identified which could potentially confound the relationships hypothesized in the model. One approach to controlling for these variables is to include them explicitly in the model. However, given the potentially large number of such variables that affect firm-level performance, the difficulty in collecting data for these variables was bound to be significant. A simpler, methodological approach to partially control for such confounds that was used in this study was random selection of respondents, a technique well established in experimental research where subjects are randomly assigned to treatments.

Findings

Because of the relatively new domain of research in which the study's constructs were being empirically examined, significant effort was directed at assessing validity and reliability. In particular, construct, external, statistical conclusion and internal validities, statistical power, and alpha reliabilities were evaluated. Model and hypotheses testing were undertaken only after a set of valid and reliable measures were established for each variable in the model (see Appendix B for these tests). The Pearson correlations between the constructs (Table 2) suggested a priori that there were many significant relationships. However, because Pearson correlations can be subject

Table 2. Correlation Matrix

	Reusability	Customizability	Process Predictability	Process Flexibility	Perceived Competitive Performance
Reusability	1.0				
Customizability	–0.09	1.0			
Process Predictability	–0.11	0.54***	1.0		
Process Flexibility	–0.08	0.39**	0.31*	1.0	
Perceived Competitive Performance	–0.02	0.49***	0.53***	0.37**	1.0

* $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$.

to spurious relationships, more accurate conclusions required path analysis, as described below.

Testing the development manager model

The first-respondent sample, consisting of responses from the VP (Development)/CTO was the basis for testing the development manager model. The theorized model fit the first-respondent data, with a p-value of chi-square (0.31) exceeding the minimum cutoff of 0.05, all three fit indices above 0.9 (NFI = 0.92, NNFI = 0.96, CFI = 0.98), and an AASR that was low (0.04). Moreover, the LM test did not suggest any additional effects to significantly increase model fit. However, the Wald test suggested that the direct effect of reusability on process flexibility and process predictability could be dropped from the model to significantly increase fit. The revised model (Figure 2), without the above two effects, had a higher fit with the data, with a p-value of chi-square of 0.50, high fit index values (NFI = 0.91, NNFI = 1.02, CFI = 1.00), and an AASR that continued to be low (0.04). Most importantly, both the LM and Wald tests suggested no further changes could be made to the revised model to significantly increase fit. The revised model—which also indicates the standardized path coefficients or effect sizes—was used to test the research hypotheses, as discussed below (Table 3).

Customizability had a significant and positive effect on process flexibility ($p < 0.005$) which suggested support for hypothesis H1, i.e., a software development process that is customizable to individual projects leads to greater flexibility in responding to changes in the product-market and regulatory environment of the firm. The standardized score for this effect was 0.40, which suggested that customizability explained about 16% of the variation in process flexibility. Customizability also had a significant and positive effect on process predictability ($p < 0.001$), which suggested support for hypothesis H2, i.e., a software development process that is customizable to individual projects results in greater predictability of the software development activities. Moreover, the standardized score for this effect was 0.54, which suggested that customizability explained over 29% of the variation in process predictability.

The Wald test suggests that effects be dropped from a model because of lack of statistical significance. As a result, the effects of reusability on process flexibility and predictability were dropped from the revised model. Model respecification therefore implied lack of support for hypotheses H3 and H4, i.e., efforts at increasing the extent of reuse of software development components did not seem to result in either greater flexibility in reacting to changes in the environment or in greater predictability of the software development activities. These surprising findings are discussed in detail in the conclusions.

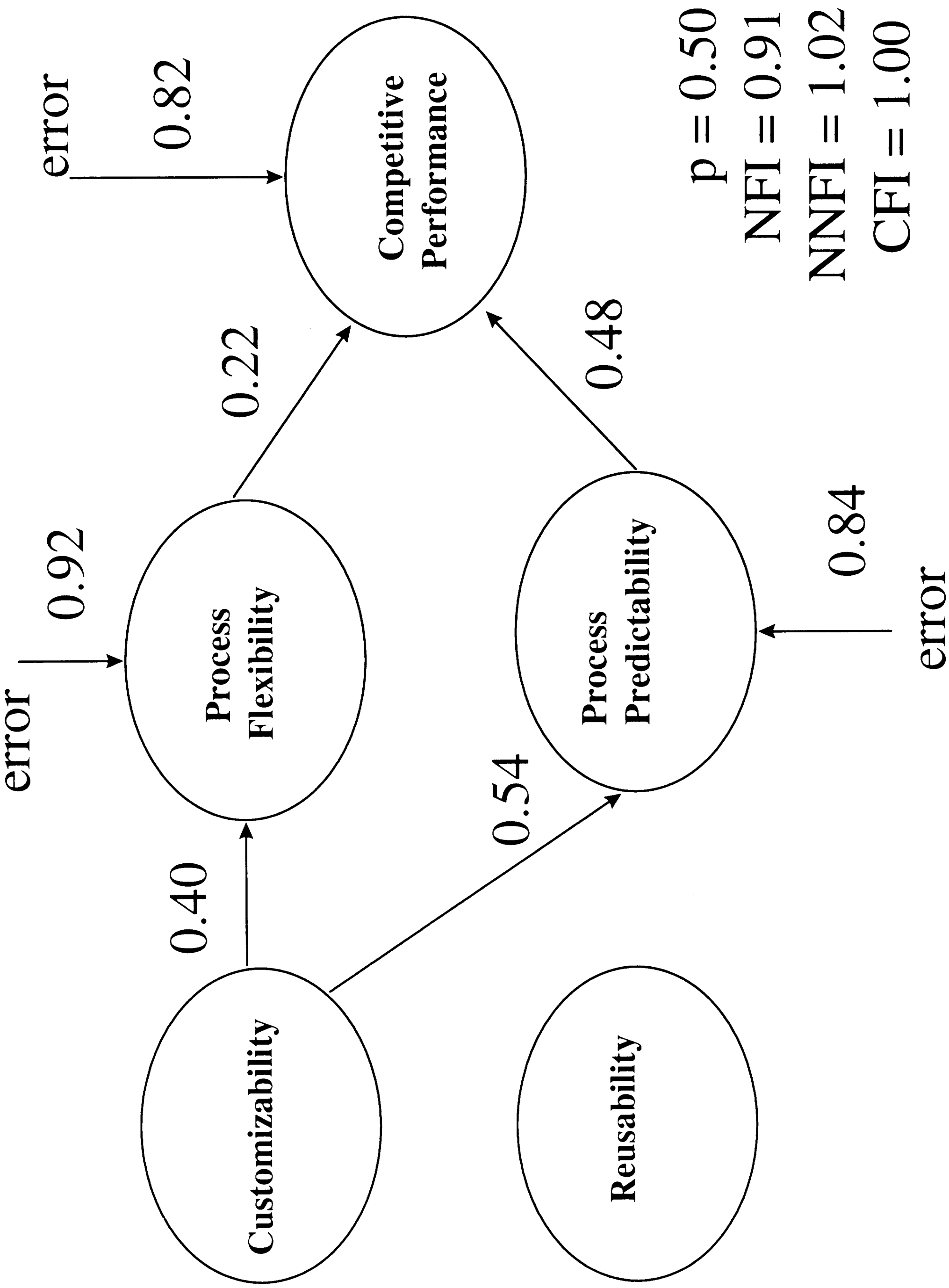

Figure 2. Revised Development Manager Model

Table 3. Perceived Performance Effects—Development Manager Model

Independent Variable	Corresponding Dependent Variable	Hypothesis	Raw Path Coefficient	Standard Error	Z-Score	Standard Path Coefficient
Customizability	Process Flexibility	H1	0.61	0.19	3.27**	0.40
	Process Predictability	H2	0.59	0.12	4.84***	0.54
Reusability	Process Flexibility	H3	Effect Dropped from Model			
	Process Predictability	H4	Effect Dropped from Model			
Process Flexibility	Perceived Competitive Performance	H5	0.23	0.11	1.98*	0.22
Process Predictability	Perceived Competitive Performance	H6	0.70	0.16	4.29***	0.48

* $p < 0.05$; ** $p < 0.005$; *** $p < 0.001$.

The direct effect of process flexibility on perceived competitive performance was also positive and significant ($p < 0.05$), which suggested support for hypothesis H5, i.e., increases in the flexibility of software development activities lead to increased perceived competitive performance of the software firm. The standardized effect was 0.22, which suggested that process flexibility explained nearly 5% of the variation in competitive performance, as perceived by the software development manager. The direct effect of process predictability on perceived competitive performance was positive and significant ($p < 0.001$), which suggested support for hypothesis H6, i.e., increases in the predictability of software development activities lead to increased perceived performance of the software firm. The standardized effect was 0.48, which suggested that process predictability explained over 23% of the variation in competitive performance, as perceived by the software development manager. In terms of actual effect sizes, results from the development manager model suggest that, except for the effect of process flexibility on perceived competitive performance (which was small-sized), the other significant effects ranged from 0.40 to 0.54 (which were close to medium-sized). The average actual effect size for all four significant effects was 0.41, which exceeded the expected effect size of 0.35. Since the actual sample size ($N = 58$) was greater than the required sample size ($N = 51$), the actual average statistical power of the findings therefore exceeded 0.80.

Testing the marketing manager model

This model was based on the sample formed by considering the VP (Marketing) assessments of competitive performance, while all other constructs were based on the VP (Development)/CTO assessments. As before, PCA was used to construct the score for the firm's perceived competitive performance, using the marketing manager's evaluation of product cost efficiency and market responsiveness. The theoretical model did not include the effects of reusability on process predictability and flexibility. These effects, which were dropped in the development manager model, would not have changed because they continued to be based on the development manager's assessments. The model now had eight parameters to be estimated. Given a sample size of 36, the sample size to parameters ratio of 4.5 was reasonably close to five, the recommended value, in order to draw meaningful conclusions from the analysis.

The theorized model fit the data, with a p-value of chi-square (0.70) considerably exceeding the minimum cutoff of 0.05, two of the three fit indices above 0.9 (NFI =0.88, NNFI = 1.17, CFI = 1.00), and an AASR that was low (0.05). Moreover, the LM test did not suggest any additional effects to significantly increase model fit. However, the Wald test suggested that the direct effect of process predictability on perceived competitive performance could be dropped from the model to significantly increase fit. The revised model (Figure 3), without the above effect, had a better fit with the data, with a p-value of chi-square of 0.80, two of the three fit indices above 0.90 (NFI = 0.88, NNFI = 1.22, CFI = 1.00), and an AASR lower than before (0.04). Most importantly, both the LM and Wald tests suggested that no further changes could be made to the revised model to significantly increase fit. The revised model—which also includes the standardized path coefficients or effect sizes—was used to test the research hypotheses, as discussed below (Table 4).

As expected, customizability had a significant positive effect on process flexibility ($p < 0.005$), with an effect size of 0.44. This reinforced support for hypothesis H1, i.e., customizable software development approaches lead to a greater ability to react effectively to changes in products, competition, or technologies. Similarly, customizability had a significant and positive effect on process predictability ($p < 0.001$), with an effect size of 0.58. This reinforced support for hypothesis H2, i.e., customizable software development approaches lead to greater predictability of the software development process. Moreover, there was no support for hypotheses H3 and H4 because the effects of reusability on process flexibility and predictability were dropped from the original model. In summary, the findings regarding H1 through H4 from a data subset (N = 36), which included marketing manager assessments of competitive performance, cross validate those from the larger set of data (N = 58).

Table 4. Perceived Performance Effects—Marketing Manager Model

Independent Variable	Corresponding Dependent Variable	Hypothesis	Raw Path Coefficient	Standard Error	Z-Score	Standard Path Coefficient
Customizability	Process Flexibility	H1	0.94	0.33	2.88**	0.44
	Process Predictability	H2	0.87	0.20	4.18***	0.58
Reusability	Process Flexibility	H3	Effect Dropped from Model			
	Process Predictability	H4	Effect Dropped from Model			
Process Flexibility	Perceived Competitive Performance	H5	0.35	0.14	2.44*	0.38
Process Predictability	Perceived Competitive Performance	H6	Effect Dropped from Model			

* $p < 0.05$; ** $p < 0.005$; *** $p < 0.001$.

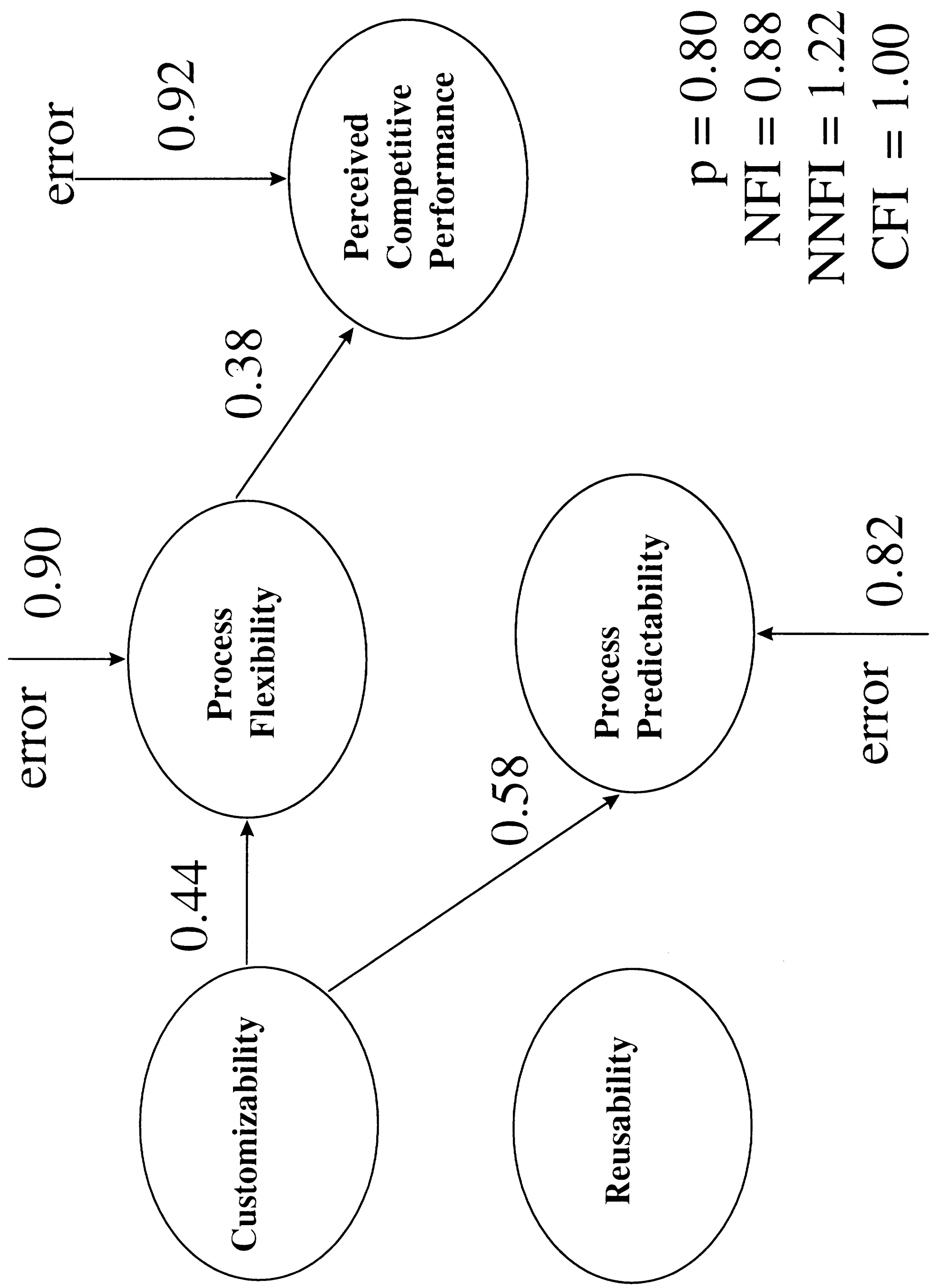

Figure 3. Revised Marketing Manager Model

The direct effect of process flexibility on perceived competitive performance was positive and significant ($p < 0.05$), which suggested support for hypothesis H5, i.e., increases in the flexibility of software development activities as perceived by the development manager were associated with increases in competitive performance as perceived by the marketing manager. Moreover, the effect size was 0.38, which suggested that process flexibility explained over 14.4% of the variation in organizational performance. However, the important difference from the development manager's model was the absence of an effect of process predictability on perceived competitive performance. This suggested lack of support for hypothesis H6, i.e., increases in process predictability as perceived by the development manager were not associated with increases in competitive performance as perceived by the marketing manager. This interesting finding is discussed in the conclusions. The average effect size for the three significant effects was 0.47, which was close to medium-sized (as expected). Moreover, the actual sample size ($N = 36$) exceeded the sample size ($N = 25$) required to obtain power levels of 0.80 (given medium-sized effects). The actual average statistical power of the findings therefore exceeded 0.80.

Internal validity

The internal validity of a model tests whether alternative explanations of the results can be provided, such as the effects of missing variables (Mitchell 1985). In this study, organizational size as a missing variable was important enough to be controlled for explicitly, while random selection of respondents was a partial methodological control for other potential confounds. For example, the relationships between the variables in the model may be more an artifact of their correlation with organizational size, rather than the presence of any effects among them. Size (given by the natural logarithm of number of full-time employees) was explicitly included by modeling its direct effect on customizability, process flexibility, and predictability in the revised development and marketing manager models. However, the results indicate lack of support for size as an explanation. The Wald test suggested dropping its effects to significantly increase model fit. Similar results were obtained when size was measured by annual sales volume.

Conclusions

Discussion of findings

Effects of customizability. The strong support for both the process customizability hypotheses (H1 and H2) suggests that customizability does have a significant influence on process predictability and flexibility, and therefore on perceived competitive performance. Specifically, firms that develop a software process that can be customized to different projects can more accurately estimate the performance of their projects, have a software development approach that can react more rapidly to changes in the environment, and therefore improve their market responsiveness and cost efficiencies vis a vis competition. In the development manager model, customizability was important enough that it had a total standardized effect of 0.35 on perceived competitive performance, therefore explaining over 12% of the latter's variation. This finding lends support to the assertions made by the SEI's CMM model that developing a customizable software development process is an important objective of sofware process maturity which leads to high-performing organizations. For the reasons described earlier, customizability ensures that there is a process template to work with and tailor to individual projects, which in turn leads to more predictable and flexible software development.

Effects of reusability. The lack of support for either of the reusability hypotheses (H3 and H4) in this study is surprising, given that process management research has emphasized reusability as an important objective. To cross validate the findings, the subjective Likert-scale measure of reusability described earlier was also considered. If we view the scale measuring percentage reuse and this subjective scale as two different methods, the MTMM matrix showed that there was very high correlation between the two methods for each kind of reusability (Table 5). Moreover, the correlations in the validity diagonal of the matrix were the highest in their corresponding row and column, suggesting good convergent and discriminant validity for the measurement scales. The study's findings were additionally supported when the subjective scale for reuse was used in subsequent analysis; the Wald test suggested that reusability's effects on process predictability and customizability should be removed in order to improve model fit.

It must be emphasized that there is no consensus among practitioners and researchers about what reuse means and how to measure it. Moreover, while reusability is based on economies of scope, it is also subject to some economies of scale, since a systematic reuse program is typically difficult and expensive to establish. Consequently, given the large number of small to medium firms in the sample, which is reflective of the software industry as a whole, it is possible that the levels of reuse were generally insufficient to affect performance significantly. At the same time, there is no way of knowing whether there would have been any relationship between reuse and the performance variables even if the levels of reuse were high. Moreover, it is possible that a philosophy of reuse imposes constraints on the process which reduces its flexibility. Overall, the lack of significant effects of reusability observed in this study and the variety of potential theoretical and methodological explanations suggest the need for considerable further research in this area.

Effects on perceived competitive performance. Recall that both the development and marketing managers rated perceived competitive performance, while process performance was rated only by the development managers. The findings suggest that, when perceived competitive performance was rated by the development managers, process predictability and flexibility were important determinants of perceived competitive performance, with more emphasis on the former dimension. However, when marketing managers rated perceived competitive performance, only process flexibility was a significant predictor.

To cross validate the above findings with the unidimensional performance scale (Dess and Robinson 1984), a correlation matrix (Table 6) was formed from combinations of respondents (development vs. marketing manager) and measurement approaches (two dimensional vs. unidimensional). The results suggest that the marketing manager had higher consistency between the two approaches to measuring performance ($r = 0.65$), when compared to the development manager ($r = 0.39$). A path analysis of the marketing manager model, based on the unidimensional scale, was next undertaken. The results provide additional support for the marketing manager's emphasis on process

Table 5. MTMM Matrix—Measurement of Reusability

	Front-End Reusability (% scale)	Back-End Reusability (% scale)	Front-End Reusability (Likert scale)	Back-End Reusability (Likert scale)
Front-End Reusability (% scale)	1.0			
Back-End Reusability (% scale)	0.54**	1.0		
Front-End Reusability (Likert scale)	0.71***	0.46**	1.0	
Back-End Reusability (Likert scale)	0.48**	0.79***	0.53***	1.0

** $p < 0.01$; *** $p < 0.001$.

Table 6. Correlation Matrix for Perceived Competitive Performance

Perceived Competitive Performance Scale		Development Manager: Two Dimensions	Development Manager: One Dimension	Marketing Manager: Two Dimensions	Marketing Manager: One Dimension
Development Manager	Two Dimensions	1.00			
	One Dimension	0.39**	1.00		
Marketing Manager	Two Dimensions	0.20	0.13	1.00	
	One Dimension	0.38*	0.01	0.65***	1.00

* $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$.

flexibility, since it had a significant and positive effect on perceived competitive performance (effect size = 0.58) and explained almost 34% of its variation. As before, the effect of process predictability on perceived competitive performance had to be dropped from the model.

The findings from Table 6 are troubling with regard to the development manager's evaluation of performance. The low correlations observed between the development manager's evaluations imply that these may have been unreliable, which could be due to single respondent bias resulting from (1) the presence of common variance leading to difficulty in conceptually distinguishing between the independent and dependent variables used in the study or (2) an intentional shading of the relationships between these variables, for example to inflate performance even if the independent variables suggested otherwise. The former explanation appears unlikely because a factor analysis of the first respondent sample with the independent and dependent variables pooled together suggested that the constructs were conceptually distinct. However, the latter explanation cannot be ruled out, which could bias the findings. Consequently, only those results that were consistent with the marketing manager's evaluations could be assumed to be reliable in this study.

When using the marketing manager's evaluations, the results suggest that increases in process flexibility, rather than process predictability, are likely to lead to increases in perceived competitive performance for the software firm. A plausible explanation for these findings may be the competitive nature of the software industry. Given the rapidly changing business and technological environment within which many software firms operate, it is possible that efforts which emphasize development of reliable estimates of costs and other resources needed for software projects may be misdirected. Instead, in such turbulent environments, successful software firms may likely be those which focus on software development processes that can respond speedily to environmental changes. In any case, the interesting, albeit mixed, findings regarding process and perceived competitive performance in this study emphasize the need for more research in this area which is sensitive to the perspective of the respondent and the competitive nature of the industry.

Limitations

There are several limitations to this study resulting from the measures used for performance, the difficulties in conducting firm-level analysis of software development practices, the lack of accepted metrics for key constructs such as reusability, and the sample size used for analysis.

First, the measurement of performance—a continuing problem in research—is a limitation of this study. One important performance-related limitation of the study was the lack of objective financial data (preferably from secondary sources such as SEC filings) for the software firms in the sample. Instead, perceptual measures of process and competitive performance had to be used in this study. Based on the 14 publicly held firms for which financial data were available, a multiple regression analysis was performed using ROI, RONA, and EPS in turn as the dependent variable, and the other constructs as independent variables. No significant coefficients were obtained for any of the effects. Since it was likely that the financial performance data may not have been normally distributed, non-parametric tests were then conducted using ranked performance scores. These tests also did not suggest any significant relationships between financial performance and the other constructs.

Another performance-related limitation was the difference in perceptions of competitive performance between the development manager and the marketing manager. In particular, the results suggest that the development managers' evaluation of competitive performance was inconsistent across the multidimensional and unidimensional scales. This unreliability poses a problem for the study, although the observed consistency in the marketing managers' evaluations implies that the problem may be due to the type of respondent, rather than the scales themselves. To reduce the impact of such respondent-related biases, only those results that were consistent across both the development manager and marketing manager samples were considered as a finding from this research.

Second, because software products and markets differ considerably among themselves, both intrafirm and interfirm differences are likely to have confounded the findings and threatened the internal validity of the research model. While this study tested explicitly for organizational size as a potential confound, random selection of firms was used to correct for the effects of the other confounds. However, random selection does not counter threats posed by covariation between the missing independent variables and those that were included in the model. Future research could undertake more explicit testing of these confounds such as the type, complexity, and size of software product.

Third, the measurement of reusability—another continuing problem in research—is a limitation of this study. Although two different measurement scales were used for cross-validation, it is likely that the interpretation of the scales may not have been uniform across respondents. It is also possible that reuse may be better measured in terms of the conditions which encourage it (such as the presence of organizational mechanisms) rather than percentage of reuse. As a check, the extent to which such mechanisms were present (see Appendix A for measurement scale) correlated significantly with interval-scaled perceptions of reuse ($r = 0.31$, $p = 0.03$), but did not correlate significantly with percentage reuse levels ($r = 0.16$, $p = 0.36$). However, it is possible that the former correlation was an artifact of the five-point Likert scales used.

Fourth, although response rates were sufficiently high in this study, the sample sizes that were used—particularly the marketing manager sample of 36 respondents—may be considered a limitation. Maximum likelihood estimation (MLE) techniques, which formed the basis for the SEM analysis, do not have good properties when sample sizes are small, which could be a possible explanation for the inconsistent findings. However, given the intense effort over two years needed to collect information for this study, attempts to generate larger samples, with high response rates, are likely to face difficulties.

Future research

The lack of support for reusability's effects, despite the attention received recently, is an opportunity for future research. Rather than concentrate on levels of reuse, researchers could investigate the performance impacts of how such reuse is implemented, e.g., whether it is planned for systematically, whether the necessary infrastructure has been put in place, and whether it is integrated into the firm's existing culture and structure. Also, the presence of organizational mechanisms, such as incentives and rewards, to encourage reuse could be critical for ensuring the success of reusability programs. Future research could also consider other dimensions of perceived competitive performance, such as product quality, or process performance dimensions, such as productivity. The selected dimensions need to be germane to the context being considered. While the study's dimensions were derived from manufacturing strategy and software engineering, a different set of theories could emphasize other dimensions.

Summary

The summarized results (Table 7) suggest that consistent support from both the development

Table 7. Summary of Findings

Hypothesis Number	Hypothesis Statement	Support From Development Manager Model?	Support From Marketing Manager Model?	Consistent Support for Hypothesis?
H1	Increases in customizability are positively associated with increases in process flexibility	Yes	Yes	Yes
H2	Increases in customizability are positively associated with increases in process predictability	Yes	Yes	Yes
H3	Increases in reusability are positively associated with increases in process flexibility	No	No	No
H4	Increases in reusability are positively associated with increases in process predictability	No	No	No
H5	Increases in process flexibility are positively associated with increases in perceived competitive performance	Yes	Yes	Yes
H6	Increases in process predictabilty are positively associated with increases in perceived competitive performance	Yes	No	No

and marketing manager models was obtained for three of the six hypotheses in the study, i.e., customizability was positively associated with process flexibility and process predictability, and process flexibility was positively associated with perceived competitive performance. There was mixed support for the hypothesis that process predictability was positively associated with perceived competitive performance, while there was no support from both models for the effects of reusability on process flexibility and process predictability. While several limitations are present, the key contribution of this study is nonetheless in developing and testing a model that relates software process engineering variables to the perceived competitive performance of the software firm. Continued research in this important area could help provide guidelines for how software firms can compete more effectively in an intensified environment.

Acknowledgements

We would like to thank Karen Stanley, Nirup Menon, and Mick McQuaid for assistance in the data collection and the Small Grants program, Office of the Vice President for Research, University of Arizona, for providing us with the funding to do this research. We would also like to thank the associate editor and three anonymous reviewers for their many comments and suggestions about this study.

References

Apte, U., Sankar, C. S., Thakur, M., and Turner, J. E. "Reusability-Based Strategy for Development of Information Systems: Implementation Experience of a Bank," *MIS Quarterly* (14:4), December 1990, pp. 421–433.

Bandinelli, S. C., Fuggetta, A., and Ghezzi, C. "Software Process Model Evolution in the SPADE Environment," *IEEE Transactions on Software Engineering* (19:12), December 1993, pp. 1128–1143.

Banker, R. D., and Kauffman, R. J. "Reuse and Productivity in Integrated Computer-Aided Software Engineering: An Empirical Study," *MIS Quarterly* (15:3), September 1991, pp. 374–399.

Banker, R. D., Chang, H., and Kemerer, C. F. "Evidence on Economies of Scale in Software Development," *Information and Software Technology* (36:5), 1994, pp. 275–282.

Bantel, K. A. "Top Team, Environment, and Performance Effects on Strategic Planning Formality," *Group and Organization Management* (18:4), December 1993, pp. 436–458.

Baroudi, J. J., and Orlikowski, W. J. "The Problem of Statistical Power in MIS Research," *MIS Quarterly* (13:1), March 1989, pp. 87–106.

Bentler, P. M. *EQS: Structural Equations Program Manual,* BMDP Statistical Software, Inc., Los Angeles, 1989.

Boynton, A. C., and Victor, B. "Beyond Flexibility: Building the Dyanamically Stable Organization," *California Management Review,* Fall 1991, pp. 53–66.

Boynton, A. C., Victor, B., and Pine, B. J. III. "New Competitive Strategies: Challenges to Organizations and Information Technology," *IBM Systems Journal* (32:1), 1993, pp. 40–64.

Carricaburu, L. "Piracy Costs Software Industry Billions," *The Salt Lake Tribune,* June 24, 1997.

Cook, T. D., and Campbell, D. T. *Quasi-Experimentation,* Houghton Mifflin Co., Boston, 1979.

Curtis, B., Kellner, M. I., and Over, J. "Process Modeling," *Communications of the ACM* (35:9), September 1992, pp. 75–90.

Cusumano, M. A. *Japan's Software Factories: A Challenge to U.S. Management,* Oxford University Press, New York, 1991.

Dess, G. B., and Robinson, R. B., Jr. "Measuring Organizational Performance in the Absence of Objective Measures: The Case of the Privately-Held Firm and Conglomerate Business Unit," *Strategic Management Journal* (5), 1984, pp. 265–273.

Dowson, M. "Software Process Themes and Issues," Second International Conference on the Software Process, Berlin, Germany, February 25-26, 1993, pp. 54–62.

Golder, P. N., and Tellis, G. J. "Pioneer Advantage: Marketing Logic or Marketing Legend?" *Journal of Marketing Research* (30), 1993, pp. 158–170.

Henderson, J. C., and Lee, S. "Managing I/S Design Teams: A Control Theories Perspective," *Management Science* (38:6), June 1992, pp. 757–777.

Hu, Q. "Evaluating Alternative Software Production Functions," *IEEE Transactions on Software Engineering* (23:6), June 1997, pp. 379–387.

Humphrey, W. S. *Managing the Software Process,* Addison-Wesley Publishing Co., Reading, MA, 1989.

Jones, T. C. "Globalization of Software Supply and Demand," *IEEE Software,* November 1994, pp. 17–23.

Jordan, W. C., and Graves, S. C. "Principles on the Benefits of Manufacturing Process Flexibility," *Management Science* (41:4), April 1995, pp. 577–594.

Keller, R. T. "Technology-Information Process Fit and the Performance of R&D Project Groups: A Test of Contingency Theory," *Academy of Management Journal* (37:1), 1994, pp.167–179.

Kotha, S. "Mass Customization: Implementing the Emerging Paradigm for Competitive Advantage," *Strategic Management Journal* (16), 1995, pp. 21–42.

Mili, H., Mili, F., and Mili, A. "Reusing Software: Issues and Research Directions," *IEEE Transactions on Software Engineering* (21:6), June 1995, pp. 528–561.

Mitchell, T. R. "An Evaluation of the Validity of Correlational Research Conducted in Organizations," *Academy of Management Review* (10:2), 1985, pp. 192–205.

Mowery, D. C. "Chapter 1: Introduction," in *The International Computer Software Industry: A Comparative Study of Industry Evolution and Structure,* D. C. Mowery (ed.), Oxford University Press, New York, 1996, pp. 3–14.

Nemetz, P. L., and Fry, L. W. "Flexible Manufacturing Organizations: Implications for Strategy Formulation and Organizational Design," *Academy of Management Review* (13:4), 1988, pp. 627–638.

Olsen, N. C. "Survival of the Fastest: Improving Service Velocity," *IEEE Software,* September 1995, pp. 28–38.

Panzar, J. C., and Willig, N. D. "Economies of Scope," *American Economic Review* (71), 1981, pp. 268–277.

Parthasarthy, R., and Sethi, S. P. "The Impact of Flexible Automation on Business Strategy and Organizational Structure," *Academy of Management Review* (17), 1992, pp. 86–111.

Paulk, M. C., Curtis, W., Chrissis, M. B., and Weber, C. V. "Capability Maturity Model, Version 1.10," *IEEE Software* (10:4), July 1993, pp. 18–27.

Pine, B. J. II. *Mass Customization: The New Frontier in Business Competition,* Harvard Business School Press, Boston, 1993.

Reifer, D. J. "Software Reuse for TQM," in *Total Quality Management for Software,* G. G. Schulmeyer and J. I. McManus (eds.), Van Nostrand Reinhold, New York, 1992, pp. 221–252.

Swanson, K., McComb, D., Smith, J., and McCubbrey, D. "The Application Software Factory: Applying Total Quality Techniques to Systems Development," *MIS Quarterly* (15:4), December 1991, pp. 567–579.

About the Authors

Sarma R. Nidumolu is an assistant professor of MIS at the University of Arizona. He has a B.S. in electronics and communications engineering from the Indian Institute of Science, a P.G.D.M. from the Indian Institute of Management, Calcutta, and a Ph.D. in management from UCLA. His research interests deal with the control and coordination of software development, and the adoption and impacts of new information technologies. This research is conducted in both domestic and international settings. He has been published in *MIS Quarterly, Information Systems Research, Management Science, Communications of the ACM,* and the *Journal of Management Information Systems,* among others.

Gary W. Knotts is a Ph.D. candidate in the MIS Department at the University of Arizona. He earned a B.S. in human factors engineer-

ing at Wright State University as well as an M.S.A. from Central Michigan University and an M.B.A. from the University of Arizona. He has articles in the proceedings of the Dallas INFORMS conference, the proceedings of the thirty-first HICSS conference, and forthcoming in the *Annals of Operations Research.* His research interests include software development, project planning and scheduling, and agent technology.

Appendix A

Questionnaire Items

Important Note:

This questionnaire contains several references to "software development approach(es)." By this we mean: the process or processes used by your organization to design, create, test, and implement software products and all related end-products (e.g., documentation). A "process" may consist of, but is not limited to, performance criteria, phases, activities, methods (e.g., tools and techniques), and personnel requirements.

Perceived Competitive Performance Dimensions

Compared to your competitors, how does your organization rate on each of the following:

(1 - at the very bottom, 3 - in the middle, 5 - at the very top)

Product cost efficiency

1. Ability to produce software at low cost for current product lines
2. Ability to charge competitive prices for software for current product lines
3. The efficiency of software production for current product lines
4. The productivity of your software developers for current product lines

Market responsiveness

1. The speed of response to new customer needs
2. Ability to tailor software products to individual customer needs
3. The speed at which new software markets can be entered
4. The rate of introduction of new software products/services

Subjective overall performance scale (Dess and Robinson 1984)

1. Overall competitive position
2. Return on assets
3. Efficiency of operations

4. Overall financial performance
5. Growth rate

Process Flexibility

Please rate the speed with which your current software development approach(es) can be used to respond effectively to the following compared to your competitors in the same situation:

(1 - slower than all, 2 - slower than most, 3 - in the middle, 4 - faster than most, 5- faster than all)

1. Quantity of developers in the labor market decreases
2. Quality of developers in the labor market decreases
3. Customer's software product requirements change
4. A new market for your software products opens
5. New competitors enter your software market
6. Changes in the basis of competition in your industry occur
7. New laws regulating your business are enacted
8. Laws regulating your business are repealed
9. New software technologies become available
10. New hardware technologies become available

Process Predictability

On average, how predictable are each of the following at the start of a software development project in your organization?

(1 - very unpredictable, 5 - very predictable)

1. Actual date when project will be completed
2. Actual developer-months that will be required
3. Actual budget that will be consumed by the project
4. Actual resources that will be required to complete the project
5. Actual quality of finished software
6. Actual functionality of finished software

Reusability

On average, please indicate the extent to which, and the percent of which, each of the following consists of reused components. (Note: Do not include components reused from previous versions of the same product.)

(1 - not at all, 3 - to some extent, 5 - to a great extent)

1.	Requirements specifications	1	2	3	4	5	___%
2.	Design specifications	1	2	3	4	5	___%
3.	Software code	1	2	3	4	5	___%
4.	Test data	1	2	3	4	5	___%
5.	Documentation	1	2	3	4	5	___%

Customizability

Please indicate the extent to which you agree or disagree with each of the following statements:

(1 - strongly disagree, 3 - neither agree nor disagree, 5 - strongly agree)

Our software development approach(es) are such that they

1. can be used effectively for a variety of projects
2. can be customized by individual project teams to suit their context
3. permit a range of responses to different types of requirements
4. permit a range of responses to different types of technologies
5. can be adapted to the quality goals of individual projects
6. can be adapted to the productivity goals of individual projects
7. can be adapted to the suit the resource levels available to individual projects

Mechanisms to Encourage Reuse

Please indicate the extent to which the following mechanisms have been developed in your organization to encourage reuse of the types of resources listed above:

(1 - does not exist, 3 - exists for some projects, 5 - exists for all projects)

1. Reward system for creating reusable resources
2. Reward system for reusing existing resources
3. Library of reusable resources
4. Reusability as formal part of employee performance evaluation
5. Reusability as formal part of software development approach

Appendix B

Purification of Measurement Scales

Construct Validity

Construct validity refers to the possibility that the measurement scale for a construct can be construed in terms of more than one construct (Cook and Campbell 1979, p. 59) and is particularly important to assess in survey research. Exploratory factor analysis using varimax rotation for the measurement scales of customizability and reusability pooled together showed that all the customizability items loaded on one factor (Table B1). However, the reusability items loaded on two factors: one described the percentage reusability of outputs such as requirements and design specifications, while the other described reuse of outputs such as software code, test data, and documentation. The first corresponded to reuse of outputs from the initial stages of software development and was labeled "front-end reusability" whereas the latter corresponded to outputs from later stages and was labeled "back-end reusability."

Exploratory factor analysis using varimax rotation for process predictability and flexibility pooled together (Table B2) revealed four factors, with all the predictability items loading on one factor, and the flexibility items loading on three (instead of five) factors. Further inspection of the flexibility factors suggested that the labor flexibility items loaded as predicted, as did the items measuring regulatory flexibility. However, the flexibility items which related to the software development approach's ability to respond to changes in products, market competition, and technology all loaded on the same factor, which made intuitive sense: software and hardware technologies are the core components of the products themselves and changes to them are often sources of new competition. This factor was labeled "product-market flexibility" for analysis.

Exploratory factor analysis using varimax rotation for the perceived competitive performance dimensions revealed two factors (Table B3) as expected, one dealing with product cost efficiency and the other dealing with market responsiveness. However, two items had to be deleted from the scale for market responsiveness because they loaded instead on product cost efficiency. To further validate these dimensions, a factor analysis was conducted of items for process predictability and product cost efficiency pooled together, and another with items for customizability, process flexibility, and market responsiveness items pooled together. The results suggested that these constructs were conceptually distinct from one another.

Reliability

Reliability assesses the stability of the instrument over various conditions and has traditionally been given by the Cronbach alpha coefficient which measures the internal consistency of the scale items. In this study, all the reliabilities ranged from 0.80 to 0.97 (Table B4).

External Validity

This concept refers to the extent to which the findings can be generalized to or across time, persons and settings (Cook and Campbell 1979, p. 39). Low response rates typically compromise external validity of the findings, and are a major source of bias in survey research. While this study's relatively high response rate (58%) suggests that nonresponse bias was unlikely, a number of tests to study external validity were nevertheless conducted. External validity would be enhanced if the sample itself was not systematically biased with regard to key characteristics such as organization size (number of employees, annual sales volume), years in business, number of projects completed each year, extent of outsourcing, and number of new products introduced each year.

Basic statistics (Table 1) suggest that the sample had wide variation in organizational characteristics: the number of years the firm was in business ranged from two to 100 years, organizational size ranged from two to 12,000 employees, annual sales ranged from $0 to $11 billion, the number of projects undertaken each year ranged from zero to 400 projects, the extent of outsourcing ranged from 0% to 80%, and the number of new products introduced each year ranged from zero to 122.

External validity would also be enhanced if the sample did not show any systematic bias in perceived process and competitive performance, which would happen if only the successful (or unsuccessful) firms decided to participate in the study. Further analysis suggested that the means and medians for every performance construct except for product cost efficiency were similar and close to three (the mid-point of the scale), skewness was less than two, and kurtosis was less than five, suggesting the scores were well distributed.

Statistical Conclusion Validity

This concept assesses whether there is a statistical basis for concluding that one construct covaries with another (Cook and Campbell 1979, p. 37) and is significantly influenced by sample size. The adequacy of sample size is dependent on the number of parameters to be estimated in the research model. In EQS, the parameters to be estimated are the path coefficients, the variances of the independent variables (including the error terms), and the covariances that are included in the model. Bentler (1989) suggests a sample size to number of parameters ratio of at least 5:1. In this study, the number of parameters to be estimated in the research model (Figure 1) were 11. Given a first-respondent sample size of 58, the ratio (5.3:1) exceeded the minimum.

Statistical Power

Another important issue in MIS research is assessment of statistical power, i.e., the probability that the statistical test will correctly reject a null hypothesis. In this study, alpha was set at 0.05 and power (P) at 0.80, according to convention. Moreover, MIS research has typically had small-to-medium effect sizes (Baroudi and Orlikowski 1989). Based on standard effect-size values (d_{small}=0.20, d_{medium}=0.50, d_{large}=0.80), the expected effect size was therefore set at 0.35, i.e., the mean of small and medium effect sizes. The required sample size for a 0.80 power level can then be calculated to be N = 51. If effect size is expected to be medium (d = 0.50), the required sample size is 25. Consequently, if actual effect sizes corresponded to those found in MIS research in general, this study's single-respondent sample size (N = 58) was expected to generate adequate power. Because software companies were the focus of this study, medium (rather than small to medium) effect sizes were likely, since the core "manufacturing" process in software companies is software development. This suggested that hypotheses testing using the double-respondent sample (N = 36) would also likely have power levels above 0.80.

Creating Composite Variables

Three of the constructs used in this study, i.e., reusability, process flexibility, and perceived competitive performance, were multidimensional. The score for each of these constructs was obtained by taking the first principal component from a principal components analysis (PCA) of the underlying dimensions (Table B5). PCA, rather than second-order factor analysis, was the data reduction technique to use because of the formative (instead of reflective) relationship between the dimensions and the construct, e.g., increases in the scores for any one dimension need not always result in increases in scores for the others. The first principal component for process flexibility represents the software development approach's ability to respond to changes primarily in products, competition, technologies, and regulation, rather than in labor supply/demand conditions.

Table B1. Factor Analysis—Reusability and Customizability

Measurement Item	Customizability	Back-End Reusability	Front-End Reusability
Percent reuse of requirements specs	0.04	0.30	0.93
Percent reuse of design specs	–0.02	0.26	0.94
Percent reuse of software code	0.05	0.77	0.33
Percent reuse of test data	–0.03	0.93	0.05
Percent reuse of documentation	–0.06	0.81	0.39
Can be used effectively for a variety of projects	0.91	–0.01	0.10
Can be customized by individual project teams to suit their context	0.79	–0.04	0.10
Permit a range of responses to different types of requirements	0.79	0.19	0.21
Permit a range of responses to different types of technologies	0.67	0.14	–0.06
Can be adapted to the quality goals of individual projects	0.86	–0.21	–0.23
Can be adapted to the productivity goals of individual projects	0.79	0.04	–0.09
Can be adapted to suit the resource levels of individual projects	0.85	–0.29	–0.01

Table B2. Factor Analysis—Process Predictability and Flexibility

Measurement Item	Process Predictability	Product Market Flexibility	Regulatory Flexibility	Labor Flexibility
Predictability of actual date of completion of a project	*0.83*	0.06	0.21	0.22
Predictability of actual developer-months for a project	*0.90*	0.10	0.10	0.08
Predictability of actual budget required for a project	*0.90*	0.14	–0.07	–0.03
Predictability of actual resources required to complete a project	*0.86*	0.18	–0.24	0.05
Predictability of actual quality of finished software	*0.65*	0.18	0.19	0.17
Predictability of actual functionality of finished software	*0.59*	0.17	0.33	–0.07
Flexibility when quantity of developers in the labor market decreases	0.14	–0.06	0.04	*0.89*
Flexibility when quality of developers in the labor market decreases	0.09	0.03	–0.04	*0.89*
Flexibility when customers' software product requirements change	0.19	*0.68*	0.40	–0.02
Flexibility when a new market for software products opens	0.02	*0.66*	0.21	–0.13
Flexibility when new competitors enter software market	0.12	*0.80*	0.05	–0.13
Flexibility when changes in basis of competition in industry occurs	0.11	*0.61*	0.24	–0.04
Flexibility when new laws regulating business are enacted	0.07	0.15	*0.92*	0.04
Flexibility when laws regulating business are repealed	0.12	0.12	*0.92*	–0.02
Flexibility when new software technologies become available	0.27	*0.80*	–0.08	0.21
Flexibility when new hardware technologies become available	0.13	*0.82*	–0.16	0.13

Table B3. Factor Analysis—Perceived Competitive Performance Dimensions

Measurement Item	Product Cost Efficiency	Market Responsiveness
Ability to produce software at low cost for current product lines	0.72	–0.04
Ability to charge competitive prices for software for current product lines	0.62	0.30
The efficiency of software production for current product lines	0.79	0.42
The productivity of software developers for current product lines	0.72	0.39
The speed of response to new customer needs*	0.85	0.16
Ability to tailor software products to individual customer needs*	0.60	0.17
The speed at which new software markets can be entered	0.17	0.90
The rate of introduction of new software products	0.21	0.88

*Deleted from subsequent analysis.

Table B4. Basic Information on Constructs and Dimensions

Construct	Dimension	Number of Items	Mean	Standard Deviation	Cronbach Alpha
Customizability		6	3.7	0.76	0.88
Reusability	Front-End Reusability (%)	2	22.0	23.5	0.97
	Back-End Reusability (%)	3	32.6	22.4	0.83
Process Flexibility	Labor Flexibility	2	3.2	0.72	0.80
	Product-Market Flexibility	6	3.4	0.59	0.84
	Regulatory Flexibility	2	3.3	0.72	0.91
Process Predictability		6	3.1	0.84	0.88
Perceived Competitive Performance	Product Cost Efficiency	4	3.7	0.68	0.81
	Market Responsiveness	2	3.0	0.76	0.83

Table B5. Principal Components Analysis

Principal Component	Percentage Variance Explained	Component Variables	Weight
Reusability	77%	Front-End Reusability	0.71
		Back-End Reusability	0.71
Process Flexibility	42.6%	Labor Flexibility	0.01
		Product-Market Flexibility	0.71
		Regulatory Flexibility	0.71
Competitive Performance	76.3%	Product Cost Efficiency	0.71
		Market Responsiveness	0.71