
Empirical Software Engineering, 6, 207–229, 2001.


# Modelling the Likelihood of Software Process Improvement: An Exploratory Study

KHALED EL-EMAM khaled.el-emam@iit.nrc.ca
*National Research Council, Montreal Road, Bldg. M-50, Ottawa, Ontario, Canada K1A OR6*

DENNIS GOLDENSON AND JAMES MCCURLEY dg@sei.cmu.edu, jmccurle@sei.cmu.edu
*Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890, USA*

JAMES HERBSLEB herbsleb@research.bell-labs.com
*Software Production Research Department, Lucent Technologies, Naperville, IL, USA*

**Abstract.** Software process assessments have become big business worldwide. They can be a powerful tool for initiating and sustaining software process improvement (SPI). However, SPI programs sometimes fail. Moreover there still are very few systematic empirical investigations about the conditions under which SPI initiatives vary in their outcomes. In this paper we present the results of a study of factors that influence the success of SPI. The data come from a sample survey of organizations that have performed assessments based on the capability maturity model for software, and was conducted from 1 to 3 years after the assessments (sufficient time had passed for changes to have taken place). The results consist of a multivariate model of the conditions (e.g., how the improvement efforts are organized and funded) that can explain the successes and failures of SPI efforts. The model is constructed using a classification tree algorithm. It identifies the most important factors that affect the outcome of SPI efforts, and describes how those factors interact with each other to influence success or failure.

**Keywords:** Software process, software process improvement, software process assessment, subjective measurement, classification trees, survey, SW-CMM

## 1. Introduction

Software process improvement (SPI) has become big business worldwide. Based in part on roots in the total quality movement following World War II, the first SEI-assisted assessment was conducted by the Software Engineering Institute (SEI) in 1987 (Humphrey and Sweet, 1987). By now thousands of organizations have been formally assessed either as part of competitions for source selection and/or by internal initiatives to improve their own organizational process capabilities.

Software process assessments can be a powerful tool for initiating and sustaining SPI. There is in fact a growing body of evidence that improving process capabilities can pay off substantially in product quality and business value (Deephouse et al., 1995–1996; El-Emam and Birk, 2000a, 2000b; Goldenson and Herbsleb, 1995; Goldenson et al., 1999; Herbsleb and Goldenson, 1996; Herbsleb et al., 1994; Herbsleb et al., 1997; Jones, 1999; Krasner, 1999; Lawlis et al., 1995).

However, evidence also shows that SPI programs sometimes fail. Two previous empirical studies have systematically investigated the factors that affect the success of SPI efforts. The first was a follow-up survey of organizations that have conducted capability maturity model (CMM)-based assessments Goldenson and Herbsleb, 1995; Herbsleb and Goldenson, 1996). The second was a follow-up study of organizations that have conducted assessments using the emerging ISO/IEC 15504 international standard (El-Emam et al., 1999).

These previous studies were limited to bivariate analysis of the factors affecting SPI success. This means that potential interactions amongst the explanatory variables were not considered. An evaluation of interaction effects is important because this may substantially amplify prerequisites to success or failure that were not detectable in previous analyses, as well as leading to a more general model of critical success factors for SPI.

In this paper we present an empirically derived model of the conditions (e.g., organization and funding of improvement efforts) that can explain the successes and failures of SPI efforts. The data set comes from a survey of organizations that have conducted CMM-based assessments (Goldenson and Herbsleb, 1995). In constructing the model we make a number of methodological and substantive contributions to the previous study in (Goldenson and Herbsleb, 1995), including that our model is multivariate, taking into account interactions, and we provide a prioritization of the factors that affect SPI success. The model is constructed using a classification tree algorithm which identifies the contribution of factors that affect the outcome of SPI efforts, and describes how those factors interact with each other to influence success or failure.

Briefly, our results indicate that the most important factor in distinguishing between success and failure of SPI efforts is the extent to which the organization is focused in its improvement effort, with clearly defined goals and consistent directions set by senior management. However we also found that the impact of this focus in SPI initiatives depends on how it is combined with organizational commitment to process improvement and with the existence of organizational politics.

The implications of these results for SPI practitioners are that they provide actionable guidance on the *most important* issues that need to be managed during an SPI effort. This guidance consists of a prioritized list of factors affecting SPI success, and a model that identifies how the most critical factors affect SPI success, and under what conditions. For researchers in this area, we provide an operational model of SPI success that can be further empirically tested and refined.

Section 2 reviews previous work and describes the general model that we test during our study. In Section 3, we present our research method, including the source of the data and the analysis methods we use. This is followed in Section 4 with the detailed results and their interpretation. We conclude the paper in Section 5 by summarizing the analytical results and providing directions for future work.

## 2. Background

There is no shortage of expert opinion on how to ensure the success of an SPI effort (Fowler and Rifkin, 1990; Maher and Gremba, 1994; McFeelay, 1996; Miller and Goldenson, 1992; Stelzer and Mellis, 1998). A recent example of success factors for a budding SPI effort has been presented by Dion (Dion, 1999):

- The need for ongoing sponsorship.
- Clear and appropriate assignment of responsibility for SPI.
- Appropriate infrastructure to support SPI (e.g., setting up a management steering group and an SEPG[1]).
- Adequate funding.
- Focusing on the needs of software projects.
- Keeping the SPI effort simple.
- Appropriate transition strategy to affect organisational change.
- Planning and monitoring of the SPI effort.
- Adequate and continuing SPI program justification.

While many of these factors have merit, it is as yet unclear which of these are the most important factors affecting SPI success, and how these factors interact to affect SPI success. It is through systematic empirical investigation that we can begin to identify the most important success factors and how they influence the success of SPI efforts.

### *2.1. Previous Studies*

Two previous empirical studies have systematically investigated the factors that affect the success of SPI efforts.[2] The first was a follow-up survey of organizations that have conducted CMM-based assessments (Goldenson and Herbsleb, 1995; Herbsleb and Goldenson, 1996). The second was a follow-up study of organizations that have conducted assessments using the emerging ISO/IEC 15504 international standard (El-Emam et al., 1999).

The CMM-based assessment study considered the relationship between a number of factors characterising the organisation and the SPI effort itself, and the organisation's success in process improvement. All of the relationships studied were bivariate. The factors that had a statistically significant association with SPI success are summarized in Table 1.

The second study also considered bivariate relationships only (El-Emam et al., 1999). It was conducted as a follow-up survey of organisations that performed assessments using the emerging ISO/IEC 15504 international standard. The analysis

*Table 1.* Factors that were found to be related to SPI success in (Goldenson and Herbsleb, 1995) based on statistical significance criteria.

| |
|---|
| Senior management monitoring of SPI |
| Compensated SPI responsibilities |
| SPI goals well understood |
| Technical staff involved in SPI |
| SPI people well respected |
| Staff time/resources dedicated to process improvement |
| Discouragement about SPI prospects |
| SPI gets in the way of ''Real'' work |
| ''Turf guarding'' inhibits SPI |
| Existence of organizational politics |
| Assessment recommendations too ambitious |
| Need guidance about how to improve |
| Need more mentoring and assistance |

identified factors directly affecting SPI success, and others that did so indirectly through another variable, namely the extent to which the SPI effort was determined by the findings and the recommendations of the assessment. These are summarized in Table 2 and Table 3 respectively.

Since the second study had a smaller scale than the first (i.e., the sample size was smaller), it is not surprising that less variables were found to be statistically significant. While there is mostly consistency in the results of the two studies, there is also a marked inconsistency. The ''organizational politics'' and the ''ambitious recommendations'' variables are statistically significant in both studies, but in the opposite direction. Specifically, in (El-Emam et al., 1999) it was found that increases in politics and ambitious recommendations were associated with greater SPI success, and the converse was found in (Goldenson and Herbsleb, 1995). There are two possible explanations for this, one substantive and one methodological.

*Table 2.* Factors that were found to be directly related to SPI success in (El-Emam et al., 1999) based on statistical significance criteria.

| |
|---|
| Ensuring that SPI goals are well understood |
| Technical staff involvement in SPI |
| Creating process action teams |

*Table 3.* Factors that were found to be indirectly related to SPI success in (El-Emam et al., 1999) based on statistical significance criteria.

| |
|---|
| Senior management monitoring of SPI |
| Compensated SPI responsibilities |
| Staff and time resources being made available for SPI |
| SPI people are well respected |
| Organizational politics |
| Ambitious recommendations from the assessment |

It is plausible that there was a cultural difference between the organisations in both data sets (the second study was conducted mainly with European organisations, while the first mainly with North American organisations). It was posited in (El-Emam et al., 1999) that organisational politics may prevent an organisation from straying too far beyond the findings of an assessment, and hence ensuring that the assessment findings are addressed. Furthermore, that the ambitious recommendations help to drive an organisation to work harder to address them, hence increasing the likelihood of SPI success.

The methodological explanation is that there is an interaction amongst these variables and other variables that is not taken into account in a bivariate analysis. Hence a multivariate analysis may help to better understand the effects.

The purpose of our study is therefore to perform a multivariate analysis of factors that affect SPI success. This would allow the identification of interactions. In addition, some of the factors that were not found to be important in these previous studies may play an important role in such interactions.

### *2.2. Model Specification*

The general model that we test is graphically represented in Figure 1. In this model, we hypothesize that there are two classes of independent variables that influence SPI success: organizational factors and process factors. We also expect there to be interactions among the two classes of independent variables.

Organizational factors are those variables that characterize the organization undergoing SPI, and the characteristics of the organizational SPI effort itself. The variables we selected for analysis are summarized in Table 4. These represent the types of organizational issues that are commonly recommended for ensuring a successful process improvement effort (Dion, 1999; Fowler and Rifkin, 1990; Maher and Gremba, 1994; McFeeley, 1996; Miller and Goldenson, 1992).

Process factors comprise those variables that characterize activities or infrastructure that are believed to be necessary for a successful SPI effort. Process

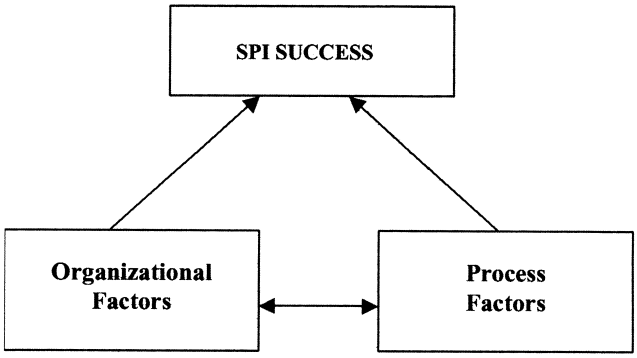



*Figure 1.* Overall specification of the model being tested.

*Table 4*. Organization questions. The response categories for these questions were "Substantial", "Moderate", "Some", and "Little if Any".

| Variable | Question |
|---|---|
| ORG1 | Are there tangible incentives or rewards for successful software process improvement? |
| ORG2 | How much does "turf guarding" inhibit the progress of software process improvement? |
| ORG3 | Is there much organizational politics? |
| ORG4 | Does senior management actively monitor the progress of software process improvement? |
| ORG5 | Is there a feeling among the technical staff that process improvement gets in the way of the real work? |
| ORG6 | To what extent are process improvement goals clearly stated and well understood? |
| ORG7 | How would you characterize the organization's staff time/resources dedicated to process improvement since the appraisal?[3] |
| ORG8 | Has there been turnover in key senior management? |
| ORG9 | Has there been involvement of technical staff in the process improvement effort? |
| ORG10 | Have the people who are involved in process improvement been respected for their technical and management knowledge, and their ability to get things done? |
| ORG11 | Has there been clear, compensated assignment of responsibilities for process improvement? |
| ORG12 | Has there been a major reorganization(s) or staff down-sizing? |
| ORG13 | How much turnover has there been among middle management? |
| ORG14 | How much turnover has there been among the technical staff? |

variables included in our analysis are summarized in Table 5. Explicit action planning (PROC1) is often said to provide a necessary basis for process improvement after an assessment (Puffer, 1999). The establishment of a functioning SEPG (PROC3) is strongly recommended for ensuring that there is an infrastructure to support SPI (Fowler and Rifkin, 1990; McFeeley, 1996). It has also been suggested that having an SEPG in a parent organization supports the functioning of an SEPG in the organization being assessed (PROC4).

## 3. Research Method

The data we use in our study were originally collected in a sample survey of organizations that conducted assessments based on the capability maturity model for

*Table 5*. Process questions. These were stated in reference to the period since the assessment has been completed. The response categories for all questions were "yes" and "no".

| Variable | Question |
|---|---|
| PROC1 | Did the organization that was assessed create an action plan for improving its software process based on the results of the assessment? |
| PROC2 | Were Process Action Teams (PATs) or similar working groups established as a result of the assessment to address specific process improvements? |
| PROC3 | Does the organization that was assessed have a software engineering process group (SEPG), or other unit(s) that performs similar functions? |
| PROC4 | Does the parent organization of the organization that was assessed have a software engineering process group (SEPG), or other unit(s) that performs similar functions? |

software (CMM®) (Goldenson and Herbsleb, 1995; Herbsleb and Goldenson, 1996). The original analysis relied largely on univariate and first-order bivariate statistics. Here we construct a multivariate model that takes into account interactions among several explanatory factors. Such a multivariate model can provide additional insights into the factors affecting SPI success beyond the original univariate and bivariate results. Furthermore, in our current analysis we prioritize these factors in terms of their contributions to SPI success.

### *3.1. Data Source*

The survey took place 1–3 years after the assessments were conducted, allowing sufficient time for changes to occur yet recent enough to expect accurate recall from the respondents. The organizations vary considerably in size and are from a wide variety of sectors and domains ranging from embedded military systems through commercial MIS.

The sample was drawn in September 1994 from the SEI's Process Appraisal Information System (PAIS) database. At that time data were available on 155 assessments in the PAIS database which met the geographic and time criteria. These assessments were conducted in the USA and Canada during calendar years 1992 and 1993. As described more fully in (Goldenson and Herbsleb, 1995) not all of the original points of contact from the database were easily accessible, and obtaining individual contact information was sometimes difficult. However, there is no a priori reason to expect any bias in the sample of 61 appraisals that were selected. This sample constitutes slightly less than 40% of the assessments in the database.

In particular, the assessments in the sample do not appear to be self-selected. Moreover, the survey respondents report widely varying degrees of success in their SPI efforts subsequent to their assessments. Even if the organizations included in our analysis are somehow more successful than others in their SPI efforts, there would have to be very substantial bias in the sample to invalidate our basic results. All told, we received completed questionnaires from 138 respondents (83% of those sent, representing 92% of the organizations that we sampled).

### *3.2. Roles of Respondents*

People who fill different roles in an organization may differ in their perspective about the same events. Hence we purposefully constructed the sample to include individuals who might be expected to differ as a result of their roles in the organization: (1) the project level software manager most knowledgeable about the appraisal; (2) the most knowledgeable and well-respected senior developer or similar technical person; (3) an organizational level SEPG manager, or someone with equivalent responsibilities for SPI.

Contrary to our original expectations, we did not find any consistent role differences among the survey respondents. We found only two statistically significant relationships ($p < 0.05$ using a $\chi^2$ test) among all of the survey questions—far fewer than would be expected by chance alone. We therefore consider that there are no role-specific differences in the responses to the questionnaire.

### *3.3. Measurement*

Our independent variables were defined as shown in Table 4 and Table 5.

We used a single survey question as the dependent variable for our analysis: ''How successfully have the findings and recommendations of the assessment been addressed?'' The response categories were ''Little If Any'', ''Limited'', ''Moderate'', ''Substantial'', and ''Marked''.

### *3.4. Missing Data Imputation*

For most organizations in our data set, there were multiple responses. We cannot simply pool all of the 138 responses received and treat this as our data set for two reasons. First, the unit of analysis is the organization rather than the individual, so the analysis should be performed at the same unit about which we wish to draw conclusions. Second, pooling responses from different roles is inappropriate because doing so would give more weight to some organization's responses relative to the others (because there are more respondents from some organizations than others), and could therefore bias the results in their favor.

Since there were no apparent differences in response patterns among the three different roles (see (Goldenson and Herbsleb, 1995)), it would be possible to analyse the data by role (i.e., divide the data set into three different subsets). The above approach, however, presents a number of difficulties. First, for some roles there are more responses than for other roles. Performing an analysis by role would result in quite small data sets for some of the role-specific subsets. Second, there existed missing values in this survey[4] (as in most surveys). A complete-case analysis by each role would reduce our number of observations considerably (see (Little and Rubin, 1987) for a discussion of complete-case analysis). Both of these difficulties can be considered as missing data problems.

One approach that can be followed to deal with missing data is based on the hot-deck imputation strategy (Sande, 1983). Hot-deck imputation has been commonly used by the US Bureau of the Census and Statistics Canada to deal with missing data problems. The basic idea of the hot-deck is to find a donor observation for each missing value. A donor observation without a missing value is identified, and the missing value is imputed from the donor. There are multiple ways in which a donor can be identified, but in our context it would be an observation from the same organization (i.e., another role).[5] For example, if the response on a question for the

SEPG manager is missing, then the response for the project manager is used. If multiple donors are identified, then one is selected at random. In practice, an observation with missing values on multiple variables may have imputed values from multiple donors (i.e., more than one role).

Hot-deck imputation, however, is based on the premise that the donor will also be used in the same analysis. This would imply that we are pooling the role-specific subsets. Since we are not doing so, an alternative strategy is employed.

The method of "field substitution" (Chapman, 1983) reserves a number of units not originally selected for sampling to replace nonresponding units. This substitute unit should have similar characteristics to the nonresponding unit. When there are multiple potential substitutes, one is selected at random. In our case, the substitute would be a respondent from a different role within the same organization. Following this approach in our data set to impute missing items, the total number of organizations for which we have complete observations was 50.[6]

### *3.5. Data Analysis Methods*

We employ two different data analysis techniques: principal components analysis and classification trees. Principal components analysis (PCA) (Kim and Mueller, 1978) is used to identify a reduced set of organizational components relative to the original set of variables. Intuitively, it would seem that some of these variables are measuring the same construct, and therefore should be combined into one composite dimension.[7] PCA provides us a systematic way for performing this reduction. Note that we are following an exploratory strategy here, as opposed to a confirmatory one since the specific dimensions are not specified a priori.

When using PCA, it is important to be able to interpret how "good" the obtained factor loadings are. Comrey (Comrey, 1973) provides some interpretation guidelines: 0.45 would be considered fair, more than 0.55 is good, those of 0.63 is very good, and those of 0.71 are excellent. We will use the 0.63 as the cutoff value in our study given the moderate sample size and its exploratory nature.

The algorithm that we used to construct classification trees was CART (Classification and Regression Trees) (Breiman et al., 1984). Constructing classification trees with CART requires that the dependent variable be discrete. We therefore dichotomized the dependent variable around the median value, differentiating between "low" success organizations and "high" success organizations. Classification tree algorithms have a number of analytical advantages. First, they do not require a detailed specification of the model to be tested beforehand, making them suitable for exploratory analysis. This is most appropriate in the formative stages of research where detailed and testable theories do not yet exist. Second, the tree is a visually interpretable structure, making the results more accessible to nonspecialists in data analysis. Third, the tree construction process takes into account potentially complex interactions among the independent variables. Fourth, it can easily accommodate categorical and continuous independent variables.

The tree construction process starts off by building a large tree, and then proceeds to prune it from the bottom up for simplification. The CART algorithm constructs binary trees. Tree-building involves the recursive partitioning of the data set. A splitting criterion is used to decide on which independent variable to split and where to make the split in the case of continuous independent variables. An exhaustive search for a "good" split is performed. For example, for a $k$ value ordered categorical variable (which all our independent variables are), there are $k-1$ possible positions to make a split.

There are a number of different ways in which the "goodness" of a split can be judged. In practice, the literature suggests that not much difference exists between the commonly used splitting criteria in terms of the accuracy of the tree that is constructed (Breiman et al., 1984; Mingers, 1989).

The splitting criterion we use is the Gini measure of node impurity (Breiman et al., 1984). The Gini measure reaches a value of zero when only one dependent variable class is present at a node. This measure is computed as the sum of products of all pairs of class proportions for classes present at the node. It reaches its maximum value when class sizes at the node are equal. Tree construction stops when a minimal number of observations in a terminal node has been reached. [8] In the present analysis, we set this minimal number at 5. Pruning starts when all terminal nodes satisfy this criterion.

During pruning, trees are generated by removing splits upward until the tree with only a root node is formed. The optimal tree can be selected from this sequence of trees. An initial choice of an optimal tree is often the one that has the highest cross-validation accuracy (we use threefold cross-validation).

However, the accuracy estimate has some uncertainty associated with it. Therefore, the optimal tree is the smallest tree (with the smallest number of terminal nodes) that is within one standard error of the smallest misclassification rate[9] (this is also referred to as the SE rule). This approach results in a tradeoff between tree accuracy and tree complexity.

After a tree is constructed, it has to be judged whether it is good or bad. A commonly used criterion for evaluating a classification model is it's overall classification accuracy (e.g., see Lanubile and Visaggio, 1997; Schneidewind, 1994). Consider the matrix in Table 6 which is commonly used to tabulate the classification results for a binary classifier. The classification accuracy is defined as:

$$\frac{n_{11}+n_{22}}{N} \tag{1}$$

A number of different approaches can be used for estimating classification accuracy. The easiest method is to use the resubstitution estimate, whereby the accuracy of the tree is evaluated using the same data set that was used to construct it. This technique, however, is known to be an optimistic estimate of accuracy on unseen cases (Weiss and Kulikowski, 1991). However, one can argue that it is a measure of the tree's "goodness of fit", in much the same manner as an $R^2$ value in an ordinary least

*Table 6*. Notation for a matrix that shows the predicted versus the actual classifications. The same matrix would be used whether the estimation approach was resubstitution or cross-validation.

| Actual success | Predicted success | | |
|---|---|---|---|
| | Low | High | |
| Low | $n_{11}$ | $n_{12}$ | $N_{1+}$ |
| High | $n_{21}$ | $n_{22}$ | $N_{2+}$ |
| | $N_{+1}$ | $N_{+2}$ | $N$ |

squares regression model. For estimating accuracy on unseen cases, the approach that we use is a threefold cross-validation.[10] Here, we split the sample into three disjoint subsamples. Two of these subsamples are used as a training set, with the third as the test set. Accuracy is calculated by classifying the observations in the test set. This is repeated by making each of the subsamples a test set, and the average accuracy across all test sets is used as the overall tree accuracy.

## 4. Results

### *4.1. Descriptive Summary*

The survey respondents represent a variety of software organizations. The largest single proportion (37%) are from organizations that do contract work for the US federal government. Another 22% are from the federal government and US military services. Firms selling in the commercial market are the second largest category (36%) of software organizations represented by our respondents. Another 5% fall into the "other" category.

The organizations represented in our sample vary considerably in size. Approximately one-third of the survey respondents say they come from organizations that have 200 or more software employees. Another third come from organizations that employ 70 or fewer people who are primarily engaged in software.

Firms selling products in the commercial market are smaller than those in the military and federal government; 43% of the commercial organizations have 70 or fewer software employees as opposed to only 14% of the government organizations. The government contractors vary more in size; 40% have 200 or more software employees, while 34% have 70 or fewer.

The survey respondents were roughly evenly distributed among the roles that were sampled: 31% are members of an SEPG and other process champions; approximately 34% are software managers and 34% are senior technical people. One person filled both the management and SEPG roles concurrently.

The respondents have a considerable amount of software experience. Half of them have worked on software for 16 years or more; a quarter of them have worked in the

field for 22 or more years. All but the least experienced 10% of our respondents have worked on software for 10 years or more.

### *4.2. Dimensions of Organizational Factors*

The results of the PCA are shown in Table 7. Note that for the results in this table all the variables within each factor were coded so that they are pointing in the same direction (i.e., higher score on each variable means a higher score on the factor).

The emergent factor structure is quite easily interpretable. We used a loading of 0.63 as a cutoff point (see Section 3.5.). Before going into the details of each of the factors, it is worthwhile to note that variable ORG12 (the occurrence of major reorganization(s) or staff down-sizing) was removed from further analysis since it seemed to relate to a number of factors and its interpretation was not obvious. Although the variable ORG9 did not load on one dominant factor, we retained this variable since a recent study identified it as an important determinant of SPI success (El-Emam et al., 1999). We termed this variable "INVOLVEMENT" since it captures the involvement of the technical staff in SPI. The remaining variables fall into one of the five components, and are interpreted below.

We used the emergent factors to construct composite variables. A composite was calculated as an unweighted sum of each of its component variables. This is a commonly used approach when working with subjective scales (Spector, 1992). For

*Table 7.* Results of the PCA with rotation for the "organization" type variables.

| | Factor 1 | Factor 2 | Factor 3 | Factor 4 | Factor 5 |
|---|---|---|---|---|---|
| ORG1 | **0.73** | −0.02 | 0.06 | −0.02 | 0.03 |
| ORG2 | −0.19 | 0.06 | **−0.86** | −0.08 | −0.005 |
| ORG3 | −0.09 | −0.16 | **−0.81** | 0.02 | −0.14 |
| ORG4 | **0.74** | 0.00 | 0.12 | −0.24 | 0.18 |
| ORG5 | −0.07 | 0.07 | 0.46 | 0.11 | **0.64** |
| ORG6 | 0.44 | −0.18 | 0.04 | 0.26 | **0.68** |
| ORG7 | **0.63** | 0.36 | 0.01 | 0.26 | 0.17 |
| ORG8 | 0.14 | 0.20 | 0.02 | −0.18 | **0.75** |
| ORG9 | 0.35 | 0.15 | 0.40 | 0.38 | 0.29 |
| ORG10 | 0.21 | 0.09 | 0.18 | **0.77** | 0.12 |
| ORG11 | **0.69** | 0.34 | 0.33 | 0.37 | −0.01 |
| ORG12 | −0.25 | −0.45 | 0.21 | 0.53 | −0.29 |
| ORG13 | −0.09 | **−0.79** | 0.25 | 0.12 | −0.12 |
| ORG14 | 0.00 | **−0.87** | 0.10 | −0.12 | 0.05 |

Approximately 67% of the variation is explained by these five components. This is a reasonable value given the exploratory nature of this study. The criterion for factor extraction was to stop when the eigenvalue went below one. This is the most common approach for deciding on the number of factors (Kim and Mueller, 1978). The coding scheme for all variables was 1 for the lowest values and 4 for the highest values.

each composite scale we calculate its Cronbach $\alpha$ coefficient (Cronbach, 1951), a measure commonly used to evaluate the reliability of subjective measurement scales (Carmines and Zeller, 1979). The coefficient can vary from 0 to 1 where 1 is perfect reliability and 0 is maximum unreliability. Nunnally has suggested that for the early stages of research a Cronbach $\alpha$ coefficient approaching 0.7 is acceptable (Nunnally, 1978). Given that we are at a formative stage in developing SPI theories, and as noted earlier our study was exploratory in nature, it would seem reasonable to use this as a general guideline to judge the reliability of the composite variables.[11]

Below is our interpretation of each of the five factors that were identified.

#### *4.2.1. Factor 1: Commitment*

The first factor is termed "Commitment". All the variables are concerned with the extent to which resources are made available for SPI and management's interest in SPI. These are considered as indicators of commitment to SPI. This four item measure of commitment had a Cronbach $\alpha$ coefficient of 0.718. The final composite variable was constructed to have higher values when commitment is high, and has a range from 4 to 16.

#### *4.2.2. Factor 2: Turnover*

The two questions that make up this variable concern the turnover at the middle management and technical levels within the organization. Note that turnover in senior management (variable ORG8) does not load on the same factor, and seems to be measuring a different construct. The Cronbach $\alpha$ coefficient for this two item variable is 0.65. Even though this number is not high, it is actually quite good for a variable consisting of only two items. The final composite variable was constructed to have high values when turnover is high, and has a range from 2 to 8. Thus, for example, for turnover to have the maximum value, there would have to be "substantial" turnover in middle management and the technical staff.

#### *4.2.3. Factor 3: Politics*

The third factor is clearly measuring an underlying construct of "Politics". This is a general label for politically motivated activities and incentives that may promote or hinder SPI within an organization. The Cronbach $\alpha$ coefficient for this two item variable was 0.732. The composite variable was constructed to have high values the more politics, and has a range from 2 to 8.

#### *4.2.4. Factor 4: Respect*

The fourth factor consists of a single item, and has been labeled "RESPECT". This measures the extent to which individuals involved in SPI are respected within the

organization. This variable was coded to have higher values the greater the respect, and has a range from 1 to 4.

#### *4.2.5. Factor 5: Focus*

The final factor measures the extent to which the organization is focused in its SPI efforts. Turnover in senior management detracts from this because new management often imposes new directions for the organization as a whole, with consequent effects on ongoing improvement initiatives. If staff feel that improvement initiatives (or the current one specifically) get in the way of the "real" work then the process improvement will be sacrificed when pressure builds up (e.g., deadlines). Finally, an organization cannot be focused in its SPI effort if its improvement goals are not clearly stated and understood. The Cronbach $\alpha$ coefficient for this variable was 0.62. The composite scale was coded so that higher values indicate greater focus, and has a range from 3 to 12.

### *4.3. Dependent Variable*

Figure 2 shows the overall distribution of the dependent variable. As can be seen, the median value is "moderate", and the modal response is "limited" success. Values at "moderate" or below are classified in the "Low" category; responses of "substantial" or "marked throughout the organization" are classified in the "high" success category.

### *4.4. Classification Tree*

The final tree is shown in Figure 3. We first present the accuracy results, and then interpret the tree.

#### *4.4.1. Accuracy*

The goodness of fit criterion (i.e., the resubstitution estimate of accuracy) for this tree was 92%, indicating a very good fit to the data. However, this is an optimistic estimate of accuracy on unseen cases.

There are two ways in which we can calculate the cross-validation estimate of accuracy. The first approach is a "global" cross-validation whereby the entire analysis is replicated three times, each time with one-third of the data set used as a test set. However, during each of these replications, these two-thirds of the data set are again split into three samples, and now only approximately 44% of the whole sample is used for constructing each of the trees that CART chooses from. With such small numbers, it is hardly surprising that not many trees are actually constructed and therefore the tree with the root node predominates. The tree with only a root node has an accuracy of 76%, which matches the base rate of 76% of organizations having "Low" SPI success. Hence this approach does not provide a reasonable estimate of the accuracy of the tree on unseen cases.

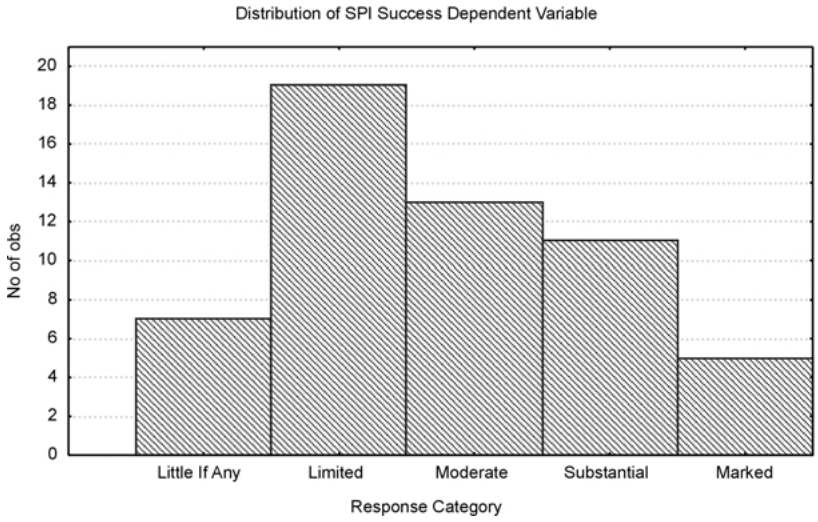



*Figure 2*. Distribution of the dependent variable "SPI success".

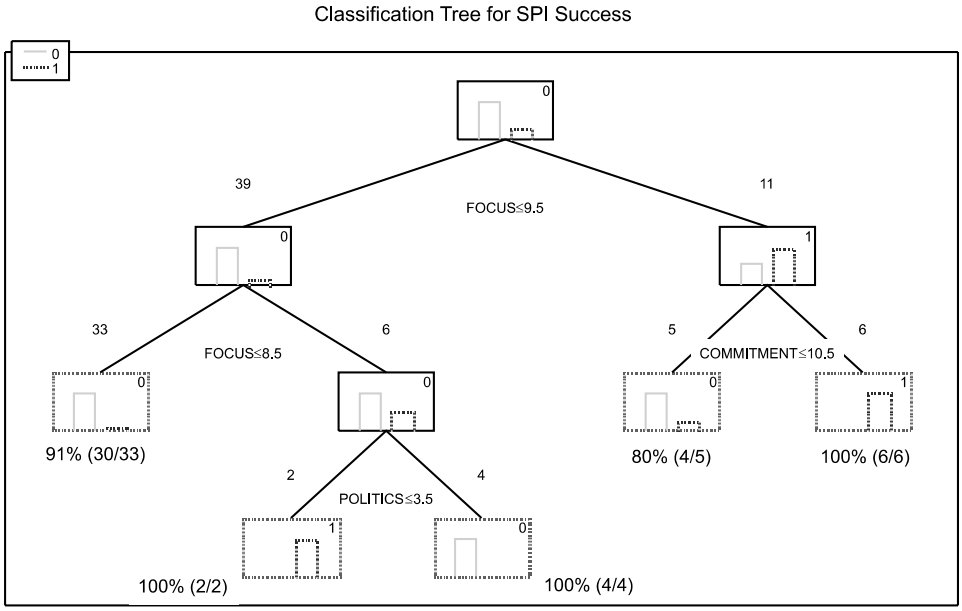



*Figure 3*. The CART generated classification tree. The number within each node is the predicted class (0 is LOW, and 1 is HIGH). The predicted class is the most frequent one within the node. The numbers on the edges are the number of observations. If the condition at a nonterminal node is true, then take the path on the left. The percentages next to each terminal node are the resubstitution estimates of accuracy for that particular terminal node.

The second approach uses the cross-validation estimate for the tree that was selected. As noted earlier, CART constructs many trees and then selects the best one. For selection of the best tree, the cross-validation estimate for each tree is used. This estimate for our tree is 84% accuracy. This too is quite high, and is a marked improvement over the chance probability of getting 50% correct classifications and the 76% base rate.

#### *4.4.2. Interpretation*

The decision tree in Figure 3 contains three variables: FOCUS, POLITICS, and COMMITMENT. Below each nonterminal node is a condition. For example, below the root node the condition is "FOCUS $<=9.5$". If this condition is true then one must traverse the left path of the tree. Otherwise traverse the right path.

All three composite variables do affect the probability that an organization will be successful in implementing process improvements based on the results of its assessment. However, the extent of such success depends on the interaction among the variables. Their effect is not interchangeable in an additive sense. High commitment and process focus pay off only when they are combined together. Organizational politics only are important when process focus is moderately high.

When process focus is low then an organization is likely to have little success in implementing process improvements based on the results of its assessment, regardless of the degree of organizational commitment or politics. Process improvement efforts are in fact more likely to be successful when there is a moderate amount of process focus, if the organization is not dominated by politics and turf guarding; otherwise the improvement effort will have little success. Implementation of process improvements is most likely to be successful if both process focus and organizational commitment are high. High amounts of process focus will not result in commensurate success when commitment remains low.[12]

The tree highlights the importance of focus and commitment as the major determinants of SPI success. Also interesting is that none of the other variables were selected in the model, indicating that their *relative* influence in explaining success is small, at least for our current sample.

### *4.5. Sensitivity of the Tree*

We investigated the sensitivity of the tree to two CART parameters: the SE rule (see Section 3.5. for a definition of the SE rule) and the minimal number of observations in a node. While some sensitivity to these parameters would be expected, the question is whether the impact is sufficient to question the stability of the results presented here.

The common default value for the SE rule that is used in selecting the optimal tree is one. Using this value means that the smallest tree within one standard error of the

smallest misclassification rate is selected. If we vary this value from 0.1 to 1.5, the same tree in Figure 3 would be selected each time. This provides reassurance that this is the optimal tree for a wide variation in this parameter.

We also varied the minimal number of observations in a node for a split to occur. Increases in the minimal value result in a tree similar to the one in Figure 3 but that is pruned further from the bottom. The reverse effect occurs when the minimal value is decreased. In both cases the cross-validation accuracy decreases, indicating that the tree in Figure 3 has the best accuracy.

### *4.6. Variable Importance*

During the tree growing process, surrogate splits are also considered. These are other splits that mimic the action of the primary splits that were actually chosen. This is evaluated using a measure of association between the alternative split and the primary split (Breiman et al., 1984). An association value of 1 indicates that the alternative split can predict perfectly the primary split.

For each variable when it appears as a surrogate, the improvements in the Gini index had that variable been selected for the primary split are summed up for all nodes. These summed improvements are scaled relative to the best performing variable such that the highest value is 100. This value is a measure of a variable's importance (Breiman et al., 1984). The importance score measures a variable's ability to mimic the chosen tree and to play a role as a stand-in for variables appearing in the primary splits.

As expected, the variables POLITICS, COMMITMENT, and FOCUS are the most important. However, the most interesting information in Figure 4 is that variables PROC1 (production of an action plan), PROC2 (establishment of PATs), PROC4 (parent organization having an SEPG), and INVOLVEMENT have little relative importance in the context of the tree shown in Figure 3. This can be interpreted to mean that in the context of our tree, these four variables do not add much to explain SPI success. Two points should be made about this assertion.

First, this does not mean that these factors are not important for SPI, only that when you consider FOCUS, COMMITMENT, and POLITICS, they have relatively much less importance. Second, this assertion is limited to our current data set. These results, however, represent an initial prioritization of the factors affecting SPI success.

## 5. Conclusions

The objective of this study was to identify a multivariate theory of SPI success. We first presented an initial high-level model that explains the success of SPI efforts based on previous work. This model consists of organizational factors and process factors that affect SPI success.

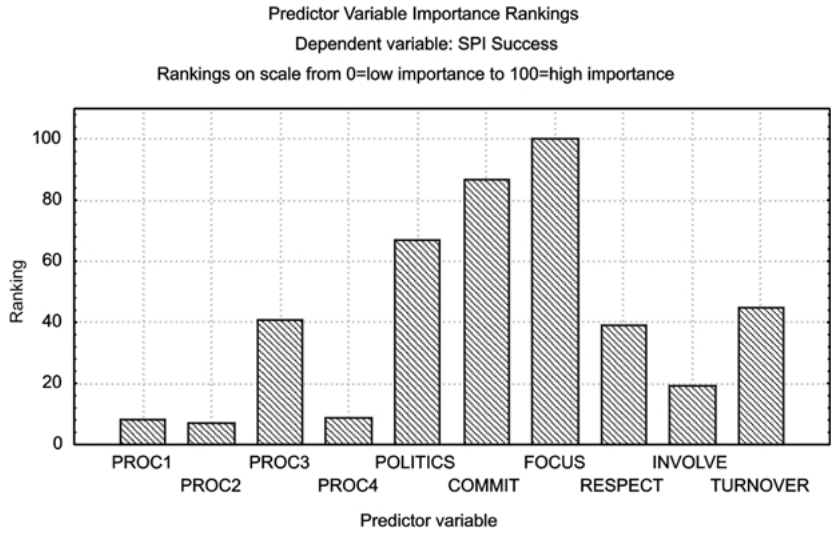



*Figure 4*. Variable importance for each of the independent variables.

Using data collected from organizations that performed CMM-based improvement, we further refined the model using an exploratory data analysis technique: classification trees. Our final classification tree had good estimated accuracy on unseen cases, and remained stable to fluctuations in the parameters of the tree building algorithm that we have used. The model provides a multivariate, empirically grounded analyses of the conditions under which process improvement is likely to succeed or fail.

Our results indicated that when an organisation is not focused in its SPI efforts (i.e., no clear SPI goals are set, SPI is not perceived as being important by the staff, and there is little continuity in management), then an organisation is likely to have little success in its SPI efforts. When the organisation is focused and commitment to SPI is high, then there is the greatest chance of SPI success. When focus is ''medium'' but there is no politics and turf guarding within the organisation, then there are also good chances of success.

The implications of our results for practitioners is that they prioritize the factors that affect SPI success, and identify a number of important interactions. This can provide useful guidance for their SPI efforts. For researchers, we have formulated a precise initial theory of SPI success. This can motivate further work in two ways. First, by providing a basis for instrument development. For instance, an instrument to measure organizational pre-requisites to successful SPI. Second, by further refining the theory through possibly the addition of more variables and/or more interactions.

Of course all conclusions from a single study are tentative at best, and require confirmation through further research. More investigation is necessary if we are to provide strongly justifiable advice to organizations on how best to conduct their improvement efforts. In particular, it would be valuable to perform a similar study

with non-CMM based improvement efforts to determine whether similar factors and interactions would be identified.

## Acknowledgments

We thank the respondents to our survey, along with those who helped construct the sample. We are grateful to our many colleagues who helped with the original study. Particular thanks are due to David White and Michael Zuccher. We also wish to thank Herb Krasner for reviewing an earlier version of this paper, as well as the anonymous reviewers of *Empirical Software Engineering* for their thoughtful comments.

## Notes

1. Software engineering process group.
2. In both studies, success was defined as the findings and recommendations of the assessment being addressed by the organisation.
3. The response categories for this question where: ''Excellent'', ''Fair'', and ''Poor''.
4. We considered a nonresponse on a question or a ''Don't Know'' response as missing values for the current study.
5. Another common way to identify donors is to define a similarity measure between observations, such as Euclidean distance. However, in our case a donor is more easily identifiable as a respondent from the same organization.
6. This is because for some organizations there were no observations at all for any role on particular variables.
7. We do not perform a PCA on the process variables because, even though they may be correlated, they do not appear to measure the same construct. Furthermore, their intercorrelations tended to be rather weak.
8. A cost complexity parameter for controlling tree growth was not used in our analysis since reduction of computation time was not a major concern.
9. It is not necessary to use the one standard error value. Later in this paper we investigate the sensitivity of our model to this value by varying this parameter.
10. We use a threefold cross-validation as an extension of the common practice of using one-third of a sample as a holdout sample for testing. Because our sample size was not very large, we refrain from using a single holdout sample approach. The threefold cross-validation also protects against fluctuations in using only one of the three subsamples for testing. The implementation of the CART algorithm is available from http://www/statsoft.com.
11. It is well known that multi-item scales are more reliable than single item scales (Zeller and Carmines, 1980). However, for some of the constructs that were measured in this study, only single item scales were used. This is the case because ours is a secondary analysis of data that was already existing, and therefore it was not possible to change the structure of the questionnaire. Also, note that the single item questions did not end up playing a role in the final set of results. This is perhaps due to their lower reliability.
12. To check for consistency across classification tree algorithms, we also constructed a tree using Quinlan's C4.5 (Quinlan, 1993). C4.5 uses an entropy based measure to decide on which variable to make the split, and an approach using confidence intervals around the estimated misclassification rate to prune the tree. The final tree that we obtained was very similar to the one shown in Figure 3 with some minor exceptions. First, the splits on the continuous variables were different. However, this is to be expected as C4.5 uses an entropy based criterion to discretize the continuous variables, which is different from the one used in CART. Also, the branch with the POLITICS variable was not taken in

the C4.5 tree. This is an indication that the entropy based splitting criterion used in C4.5 does not identify a gain in further partitioning the "medium FOCUS" node. However, the C4.5 tree had a 71% accuracy using the threefold cross-validation estimate, which is less than the CART accuracy. It is also less then the base rate of 76%. This indicates that the lack of the POLITICS split has reduced considerably the ability of the tree to predict unseen cases. We also found the C4.5 tree to be sensitive to settings of some of its parameters. By changing the minimal number of observations in a node, we obtained a tree that uses the variable PROC3 instead of COMMITMENT. This is can be considered consistent, however, as one can argue that the existence of an SEPG is a manifestation of an organization's commitment.

## References

Breiman, L., Friedman, J., Olshen, R., and Stone, C. 1984. *Classification and Regression Trees*. Wadsworth and Brooks Cole.

Carmines, E., and Zeller, R. 1979. *Reliability and Validity Assessment*. Sage Publications.

Chapman, D. 1983. *The impact of substitution on survey estimates* In: W. Madow, I. Olkin and D. Rubin (eds.): *Incomplete Data in Sample Surveys*. Vol. 2. Academic Press.

Comrey, A. 1973. *A First Course on Factor Analysis*. Academic Press.

Cronbach, L. 1951. Coefficient alpha and the internal structure of tests. *Psychomterika* 16(3): 297–334.

Deephouse, C., Mukhopadhyay, T., Goldenson, D., and Kellner, M. 1995–1996. Software processes and project performance. *Journal of Management Information Systems* 12(3).

Dion, R. 1999. Starting the climb towards the CMM level 2 plateau. In: K. El-Emam and N. H. Madhavji (eds.): *Elements of Software Process Assessment and Improvement*, IEEE CS Press.

El-Emam, K., and Birk, A. 2000a. Validating the ISO/IEC 15504 measures of software development process capability. *Journal of Systems and Software* 51(2): 119–149.

El-Emam, K., and Birk, A. 2000b. Validating the ISO/IEC 15504 measures of software requirements analysis process capability. *IEEE Transactions on Software Engineering* 26(6): 541–566.

El-Emam, K., Smith, B., and Fusaro, P. 1999. Success factors and barriers in software process improvement: An empirical study. In: R. Messnarz and C. Tully (eds.): *Better Software Practice for Business Benefit: Principles and Experiences*. IEEE CS Press.

Fowler, P., and Rifkin, S. 1990. *Software Engineering Process Group Guide*. Software Engineering Institute, CMU/SEI-90-TR-24.

Goldenson, D. R., and Herbsleb, J. 1995. *After the Appraisal: A Systematic Survey of Process Improvement, its Benefits, and Factors that Influence Success*. Software Engineering Institute, CMU/SEI-95-TR-009.

Goldenson, D., El-Emam, K., Herbsleb, J., and Deephouse, C. 1999. Empirical studies of software process assessment methods. In: K. El-Emam and N. H. Madhavji (eds.): *Elements of Software Process Assessment and Improvement*. IEEE CS Press.

Herbsleb, J., and Goldenson, D. 1996. A Systematic Survey of CMM Experience and Results. *Proceedings of the International Conference on Software Engineering*. pp. 25–30.

Herbsleb, J., Carleton, A., Rozum, J., Siegel, J., and Zubrow, D. 1994. *Benefits of CMM-based Software Process Improvement: Initial Results*. Software Engineering Institute, CMU/SEI-94-TR-13.

Herbsleb, J., Zubrow, D., Goldenson, D., Hayes, W., and Paulk, M. 1997. Software quality and the capability maturity model. *Communications of the ACM* 40(6): 30–40.

Humphrey, W., and Sweet, W. 1987. *A Method for Assessing the Software Engineering of Contractors*. Software Engineering Institute, CMU/SEI-87-TR-0023.

Jones, C. 1999. The economics of software process improvements. In: K. El-Emam and N. H. Madhavji (eds.): *Elements of Software Process Assessment and Improvement*. IEEE CS Press.

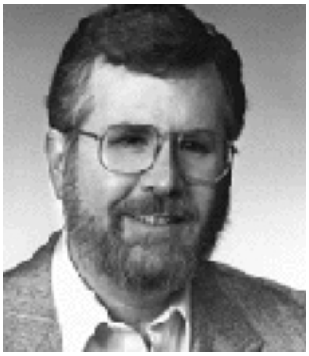
Kim, J., and Mueller, C. 1978. *Factor Analysis: Statistical Methods and Practical Issues*. Sage Publications.

Krasner, H. 1999. The payoff for software process improvement: What it is and how to get it. In: K. El-Emam and N. H. Madhavji (eds.): *Elements of Software Process Assessment and Improvement*. IEEE CS Press.

Lanubile, F., and Visaggio, G. 1997. Evaluating predictive quality models derived from software measures: Lessons learned. *Journal of Systems and Software* 38: 225–234.
Lawlis, P., Flowe, R., and Thordahl, J. 1995. A correlational study of the CMM and software development performance. *CrossTalk* 21–25.
Little, R., and Rubin, D. 1987. *Statistical Analysis With Missing Data*. John Wiley & Sons.
Maher, J., and Gremba, J. 1994. Organizational Barriers to PI and Technology Transition. *Proceedings of the 1994 SEI Software Engineering Symposium*.
McFeeley, B. 1996. *IDEAL: A User's Guide for Software Process Improvement*. Software Engineering Institute, Handbook CMU/SEI-96-HB-001.
Miller, M., and Goldenson, D. 1992. *Software Engineering Process Groups: Results of the 1992 SEPG Workshop and a First Report on SEPG Status*. Software Engineering Institute, Special Report CMU/SEI-92-SR-13.
Mingers, J. 1989. An empirical comparison of selection measures for decision-tree induction. *Machine Learning* 3: 319–342.
Nunnally, J. 1978. *Psychometric Theory*. McGraw-Hill.
Puffer, J. 1999. Action planning. In: K. El-Emam and N. H. Madhavji (eds.): *Elements of Software Process Assessment and Improvement*. IEEE CS Press.
Quinlan, J. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann.
Sande, I. 1983. Hot-deck imputation procedures. In: W. Madow and I. Olkin (eds.): *Incomplete Data in Sample Surveys*. Vol. 3. Academic Press.
Schneidewind, N. 1994. Validating metrics for ensuring space shuttle flight software quality. *IEEE Computer* 50–57.
Spector, P. 1992. *Summated Rating Scale Construction*. Sage Publications.
Stelzer, D., and Mellis, W. 1998. Success factors of organizational change in software process improvement. *Software Process: Improvement and Practice* 4(4): 227–250.
Weiss, S., and Kulikowski, C. 1991. *Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems*. Morgan Kaufmann Publishers.
Zeller, R., and Carmines, E. 1980. *Measurement in the Social Sciences*. Cambridge University Press.

**Khaled El-Emam** is currently a Research Officer at the National Research Council in Ottawa. He is co-editor of ISO's project to develop an international standard defining the software measurement process (ISO/IEC 15939), and leading the software engineering process area in the IEEE's project to define the Software Engineering Body of Knowledge. He has also co-edited two books on software process, both published by the IEEE CS Press, and is an adjunct professor at both the School of Computer Science at McGill University and the Department of Computer Science at the University of Quebec at Montreal. Previously, he was the International Trials Coordinator for the SPICE Trials, where he was leading the empirical evaluation of the emerging process assessment International Standard, ISO/IEC 15504, worldwide; the head of the Quantitative Methods Group at the Fraunhofer Institute for Experimental Software Engineering in Germany;

a research scientist at the Centre de recherche informatique de Montreal (CRIM) in Canada; a researcher in the software engineering laboratory at McGill University; and worked in a number of research and development projects for organizations such as Toshiba International Company and Honeywell Control Systems in the UK, and Yokogawa Electric in Japan. His current work focuses on object-oriented measurement, and can be found at (http://www.object-oriented.org). Khaled El Emam obtained his Ph.D. from the Department of Electrical and Electronics Engineering, King's College, the University of London (UK) in 1994.



**Dennis R. Goldenson** is a senior member of the technical staff in the Software Engineering Measurement and Analysis group at the Software Engineering Institute in Pittsburgh, Pennsylvania, USA. His work focuses on the use of measurement and analysis in software engineering, the improvement of process appraisal methods and models, and the impact of software process improvement and other software engineering practices. Related interests are in tools to support collaborative processes, survey research, and experimental design. Dr. Goldenson is a principal author of the Measurement and Analysis Process Area for CMMIsm and currently serves as co-lead of test and evaluation for the project. Additionally, he is the international trials coordinator for empirical methods of the SPICE project in support of ISO/IEC 15504. Other recent work has included requirements elicitation for training on the acquisition of software intensive systems, and a study of the practices of high maturity organizations.

**Jim McCurley** is a member of the Software Engineering Measurement and Analysis (SEMA) group at the Software Engineering Institute, Carnegie Mellon University. His current work, as a member of the Incident Analysis team in CERT, focuses on network defense.

**James D. Herbsleb** is currently a member of the Software Production Research Department, and leader of the Bell Labs Collaboratory project. He holds an M.S. in computer science from the University of Michigan, and a Ph.D. in psychology from University of Nebraska. He has held positions as a post-doctoral research fellow at the University of Michigan, and a senior member of the technical staff of the Software Engineering Institute at Carnegie Mellon University. He took his current position in Bell Laboratories Research in 1996. For the past 10 years, he has conducted research in the areas of collaborative software engineering, human–computer interaction, and computer supported co-operative work. For the last 3 years, his work has focused on collaboration technology to support large globally distributed projects.