SPICE: An Empiricist's Perspective

KHALED EL EMAM
DENNIS R. GOLDENSON

International Software Engineering Research Network Technical Report ISERN-98-03
(this report was originally written in 1995)

SPICE: An Empiricist's Perspective✧

Khaled El Emam†
Dennis R. Goldenson*

†Fraunhofer Institute for Experimental Software Engineering
*Software Engineering Institute, Carnegie Mellon University

Abstract

The SPICE project aims to deliver an international standard for software process assessment by the end of 1996. As part of this project there is an empirical trials phase whose purpose is to ascertain the effectiveness of the prospective SPICE standard. Two of the objectives of the trials phase are: (a) to determine the extent to which SPICE-conformant assessments are repeatable (i.e., reliability), and (b) to determine the extent to which SPICE-conformant assessments are really measuring best software process practices (i.e., validity). This paper introduces the theoretical foundations for evaluating the reliability and validity of measurement, suggests some empirical research methods for investigating them in SPICE, and discusses the constraints and limitations of these methods within the context of the SPICE project.

1 Introduction

Over the last two years there has been an on-going effort at developing an international standard for software process assessment. This effort is known as the SPICE (Software Process Improvement and Capability dEtermination) project. A prime motivation for developing this prospective standard has been the perceived need for an internationally recognized software process assessment framework that pulls together the existing public and proprietary methods [53]. Overviews of this project have been presented by various members of the SPICE team [21][22][23][41][52].

One important question that ought to be asked about such a prospective standard is: *"does it embody sound software engineering practices?"* This question reflects a more general concern among some researchers that existing software engineering standards lack an empirical basis demonstrating that they indeed represent "good" practices. For instance, it has been noted that [55] *"standards have codified approaches whose effectiveness has not been rigorously and scientifically demonstrated. Rather, we have too often relied on anecdote, 'gut feeling,' the opinions of experts, or even flawed research,"* and [54] *"many corporate, national and international standards are based on conventional wisdom [as opposed to empirical evidence]."* Similar arguments are made in [28][29][30].

To address such shortcomings in previous standardization efforts, the SPICE project includes an empirical trials phase. While ideally an accumulation of empirical evidence ought to precede a standardization effort, inclusion of the trials phase in the SPICE project is still a considerable improvement over previous software engineering standardization efforts.

SPICE essentially defines requirements for a measurement procedure. In other scientific disciplines (such as educational research, psychometrics, and econometrics), measurement procedures are expected to exhibit high reliability and high validity. Thus, two of the main objectives of the SPICE trials phase are: (a) to determine the extent to which SPICE-conformant assessments[1]

✧ This work was done while El Emam was at CRIM, Montreal. The views stated in this paper are those of the authors and do not necessarily reflect the policies or positions of CRIM, Fraunhofer IESE, the Software Engineering Institute, Carnegie Mellon University, or their funding agencies.

1 In SPICE, explicit conformance criteria are defined. Thus, in this paper when we write about SPICE-conformant assessments, we refer to assessments that satisfy the conformance criteria.

are repeatable (i.e., their reliability), and (b) to determine the extent to which SPICE-conformant assessments are really measuring best software process practices (i.e., their validity).

In this paper we first introduce the theoretical foundations for evaluating the reliability and validity of measurement procedures in general. We then suggest some empirical research methods for evaluating the reliability and validity of SPICE, and discuss the constraints and limitations of these methods within the context of the SPICE project.

The main contributions of this paper are: (a) to highlight the importance of the trials in ascertaining the effectiveness of SPICE, at least on scientific grounds, (b) to bring awareness of some important empirical questions that ought to be addressed by any software process assessment method, and (c) to present some theoretical concepts and empirical research methods for evaluating the reliability and validity of software process assessments.

In the remainder of this paper, each of the SPICE trials' reliability and validity objectives is discussed in a section of its own (sections 2 and 3 respectively). This is followed in section 4 by a discussion of the limitations and necessary tradeoffs in conducting the SPICE trials. Section 5 concludes the paper with a summary of the main points.

2 Reliability

2.1 The Importance of Reliability

A SPICE-conformant assessment is a measurement procedure. For any measurement procedure, reliability is of fundamental concern. Reliability addresses the extent to which there exists random error in a measurement procedure.

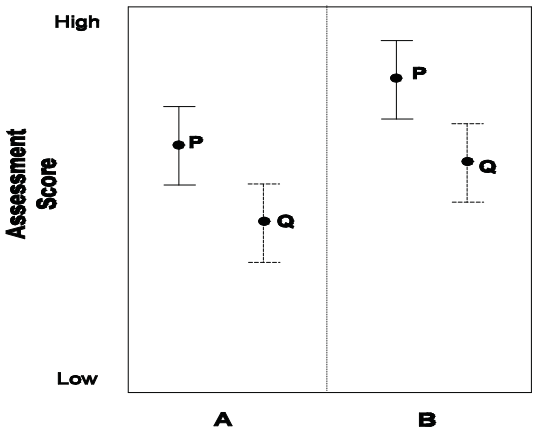

Figure 1: Example hypothetical assessment scores with confidence intervals.

Within the context of software process assessments, concern with reliability is not unique to SPICE. For example, Card broached reliability issues while discussing the repeatability of CMM-based Software Capability Evaluations [12].

Ideally, SPICE-conformant assessments should exhibit high reliability. This means that random error is minimal and that assessment scores are consistent and repeatable.

The extent of reliability of SPICE-conformant assessments and the sources of random measurement error can be ascertained empirically. Such empirical studies are being conducted during the trials phase of the SPICE standardization project.

Reliability studies in the SPICE trials are not simply an academic exercise. These studies are being designed to be in concert with the types of decisions that will be made in practice based on the outcomes of SPICE-conformant assessments.

Two common types of decisions that will be made based on SPICE-conformant assessment outcomes are considered here. These decisions are represented by the following two scenarios: (a) a contract award scenario where a contractor has to select between competing suppliers, and (b) a self-improvement scenario where an organization identifies areas for improvement and tracks improvement progress.

Assume Figure 1 shows the profiles[2] of two organizations, A and B, and that P and Q are two different processes being assessed. Due to random measurement error, the scores obtained for each process are only one of the many possible scores that would be obtained had the organization been repeatedly assessed[3]. While obtained scores for organization B are in general higher than those of organization A, this may be an artifact of chance. Without consideration of random measurement error, organization A may be unfairly penalized in a contract award situation.

2 For simplicity, the diagram in Figure 1 does not illustrate profiles in the same way as intended in the prospective SPICE standard. This, however, does not result in any loss of generality in the discussion.

3 The confidence intervals are established by adding and subtracting, for instance, one standard error of measurement to/from the obtained score after it is transformed to a z-score. The standard error of measurement can be calculated from the reliability estimate (see [51]).

Turning to a self-improvement scenario, assume that Figure 1 shows the profiles of one organization at two points in time, A and B. At time A, it may seem that the score for process Q is much lower than for process P. Thus, the organization would be tempted to pour resources on improvements in process Q. However, without consideration of random measurement error, one cannot have high confidence about the extent to which the difference between P and Q scores is an artifact of chance. Furthermore, at time B, it may seem that the organization has improved. However, without consideration of random measurement error, one cannot have high confidence about the extent to which the difference between A and B scores (for processes P and Q) are artifacts of chance.

The above examples highlight the importance of evaluating the extent of reliability of SPICE-conformant assessments. Furthermore, to facilitate improved decision-making based on such assessments, the sources of error ought to be identified, and reduced if possible, in decision making situations.

Another benefit of evaluating reliability is that one can determine how to *increase* the reliability of SPICE-conformant assessments. For example, one can estimate what the reliability coefficient *would be* if the length of an instrument is increased, or if more assessment teams conducted the assessments concurrently and aggregated their findings. This information would be useful for improving the prospective SPICE standard before standardization, as well as for providing directions to users of the eventual standard about how to conduct more reliable assessments if the context demands it.

In the appendix of this paper we have included some general guidelines for increasing the reliability of software process assessments. These guidelines should be useful for others who are developing or using software process assessment methods.

The remainder of this section includes an overview of reliability theory and methods for reliability estimation, and a discussion of their application in estimating reliability in the SPICE trials.

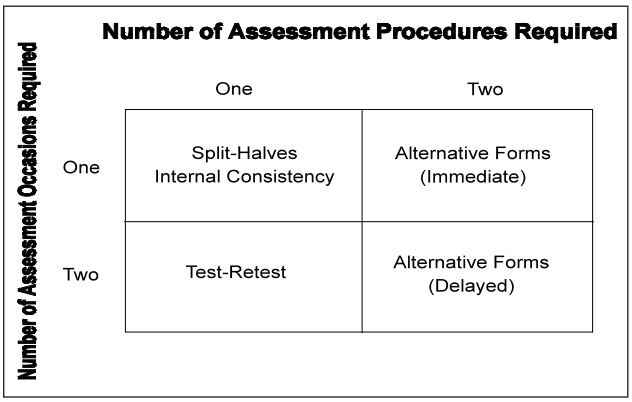

Figure 2: A classification of classical reliability estimation methods.

2.2 Reliability Theory and Methods

Two theoretical frameworks for ascertaining the extent of reliability are presented below: (a) the classical test theory framework, and (b) the generalizability theory framework. Associated with each theoretical framework are a number of empirical research methods that can be applied.

2.2.1 Classical Test Theory Framework

Classical test theory states that an observed score consists of two additive components, a true score and an error: X = T + E. Thus, X would be the score obtained in a SPICE-conformant assessment, T is the mean of the theoretical distribution of X scores that would be found in repeated assessments of the same organization[4], and E is the error component. The reliability of measurement is defined as the ratio of true score variance to observed score variance.

There are four methods for *estimating reliability* under this framework. All of the four methods attempt to determine the proportion of variance in a measurement scale that is systematic. The different methods can be classified by the number of different assessment procedures necessary and the number of different assessment occasions necessary. This classification is depicted in Figure 2. These methods are briefly described below (for more details see [45]):

1. Test-Retest Method

This is the simplest method for estimating reliability. In the SPICE context, one would have to assess each organization's capability at two points in time using the same assessment procedure (i.e., the same instrument, the same assessors, and the same assessment process). Reliability would be estimated by the correlation

4 In practice the true score can never be really known since it is generally not possible to obtain a large number of repeated assessments of the same organization. If one is willing to make some assumptions (e.g., an assumption of linearity), however, point estimates of true scores can be computed from observed scores [45].

between the scores obtained on the two assessments.

2. *Alternative Forms Method*
   Instead of using the same assessment procedure on two occasions, the alternative forms method stipulates that two alternative assessment procedures be used. This can be achieved, for example, by using two different instruments or having two alternative, but equally qualified, assessment teams. This method can be characterized either as immediate (where the two assessments are concurrent in time), or delayed (where the two assessments are separated in time). The correlation coefficient (or some other measure of association) is then used as an estimate of reliability of *either* of the alternative forms.
3. *Split-Halves Method*
   With the split-halves method, the total number of items in an assessment instrument are divided into two halves and the half-instruments are correlated to get an estimate of reliability. The halves can be considered as approximations to alternative forms. A correction must be applied to the correlation coefficient though, since that coefficient gives the reliability of each half only. One such correction is known as the Spearman-Brown prophecy formula [51].
4. *Internal Consistency Methods*
   With methods falling under this heading, one examines the covariance among all the items in an assessment instrument. By far the most commonly used internal consistency estimate is the Cronbach alpha coefficient [16].

Since there exists more than one classical method for estimating reliability, a relevant question is: *"which method(s) are most commonly reported in the literature?"* If we take the field of Management Information Systems (MIS) as a reference discipline (in the sense that MIS researchers are also concerned with software processes, their measurement, and their improvement), then some general statements can be made about the perceived relative importance of the different methods.

In MIS, researchers developing instruments for measuring software processes and their outcomes tend to report the Cronbach alpha coefficient most frequently [61]. Furthermore, some researchers consider the Cronbach alpha coefficient to be the most important [58].

Examples of instruments with reported Cronbach alpha coefficients are those for measuring user information satisfaction [38][62], user involvement [5][2], and perceived ease of use and usefulness of software [18][1]. However, recently, reliability estimates using other methods have also been reported, for example, test-retest reliability for a user information satisfaction instrument [32], and for a user involvement instrument [63].

In software engineering, the few studies that consider reliability, report the Cronbach alpha coefficient. For example, the reliability estimate for a requirements engineering success instrument [25], for an organizational maturity instrument [26], and for level 2 and 3 questions of the preliminary version of the SEI maturity questionnaire [37].

2.2.2 Generalizability Theory Framework

The different classical methods for estimating reliability presented above vary in the factors that they subsume under error variance. Figure 3 summarizes these factors. This means that the use of different classical methods will yield different estimates of reliability.

Generalizability theory [17], however, allows one to *explicitly consider multiple sources of error simultaneously and estimate their relative contributions*. In the context of SPICE, the theory would be concerned with the *accuracy of generalizing from an organization's obtained score on a SPICE-conformant assessment to the average score that the organization would have received under all possible conditions of assessment* (e.g., using all SPICE-conformant instruments, all SPICE-conformant assessment teams, all assessment team sizes, etc.). This average score is referred to as the *universe score*. All possible conditions of assessment are referred to as the *universe of assessments*. A set of measurement conditions is called a *facet*. Facets relevant to SPICE include instrument used, assessment team, and assessment team size.

Generalizability theory uses the factorial analysis of variance (ANOVA) [50] to partition an organization's assessment score into an effect for the universe score, an effect for each facet or source of error, an effect for each of their combinations, and other "random" error. This can be contrasted to simple ANOVA, which is more

analogous to the classical test theory framework. With simple ANOVA the variance is partitioned into "between" and "within". The former is thought of as systematic variance or signal. The latter is thought of as random error or noise. In the classical test theory framework one similarly partitions the total variance into true score and error score.

Suppose, for the purpose of illustration, one facet is considered, namely assessment instrument.

Source of Error	Description
Different Occasions	Assessment scores may differ across time. Instability of assessment scores may be due to temporary circumstances and/or actual change.
Different Assessors	Assessment scores may differ across assessors (or assessment teams). Lack of repeatability of assessment scores may be due to the subjectivity in the evaluations and judgement of particular assessors (i.e., do different assessors make the same judgements about an organization's processes?).
Different Instrument Contents	Assessment scores may differ across instruments. Lack of equivalence of instruments may be due to the questions in different instruments not being constructed according to the same content specifications (i.e., do different instruments have questions that cover the same content domain?).
Within Instrument Contents	Responses to different questions or subsets of questions within the same instrument may differ amongst themselves. One reason for these differences is that questions or subsets of questions may not have been constructed to the same or to consistent content specifications. Regardless of their content, questions may be formulated poorly, may be difficult to understand, may not be interpreted consistently, etc.

Figure 3(a): Definition of some sources of error.

Sources of Error \ Reliability Estimation Methods	Test-Retest	Alternative-Forms (instruments-delayed)	Alternative-Forms (instruments-immediate)	Alternative-Forms (assessors-delayed)	Alternate-Forms (assessors-immediate)	Split-Halves	Internal-Consistency
Different Occasions	X	X		X			
Different Assessors				X	X		
Different Instrument Contents		X	X				
Within Instrument Contents						X	X

Figure 3(b): Sources of error accounted for by the classical reliability estimation methods.

Further, suppose that in the trials phase two instruments are used and N organizations are assessed using each of the two instruments. In this case, one intends to generalize from the two SPICE-conformant instruments to all other SPICE-conformant instruments. This design is represented in Figure 4. The results of this study would be analyzed as a two-way ANOVA with one observation per cell (e.g., see [50]). The above example could be extended to have multiple facets (i.e., to account for multiple sources of error such as instruments *and* assessors).

2.2.3 Other Methods

Methods based on the above two frameworks are not the only ones that can be used in reliability studies. However, they are the most well developed in the literature and the most commonly used. Other methods that may provide useful information as to the consistency and repeatability of SPICE-conformant assessments include the calculation of proximity or similarity measures of SPICE-conformant assessment profiles (produced by different assessment teams), and indices of agreement [31][24].

2.3 Application to SPICE

A number of reliability studies are being planned during the SPICE trials. One primary purpose of these studies is to estimate the reliability of SPICE-conformant assessments. Evidence of good reliability would commonly constitute coefficients of at least 0.8, but preferably 0.9 or higher.

Given the complexity of an intervention such as a SPICE-conformant assessment (or a software process assessment in general), only a subset of those methods presented earlier are feasible. The most feasible classical approaches for assessing

Organization	Instrument 1	Instrument 2
1		
2		
3		
4		
5		
... ...		
N		

Figure 4: A basic one facet design.

reliability are: alternative forms (immediate) with different assessment teams or different assessment instruments, split-halves, and possibly internal consistency.

The alternative forms method with different assessment teams concerns the issue of inter-assessor reliability. This source of error is perhaps the one of most concern to the software process community. There are two general approaches that can be utilized to investigate this kind of error.

The first approach would be in a 'lab' setting. Fortunately, lab settings exist whenever a number of assessors take SPICE assessment courses or briefings as part of their qualification or training. Realistic case studies could be given to the assessors and they would be requested to provide their ratings. Individual (or team) ratings would be compared across assessors (or teams) to determine reliability.

The second approach would be in a 'field' setting. For example, assessment teams could be divided into two groups. Each group would perform its rating independently and subsequently meet to arrive at a consensus on the final ratings. The independent ratings would be compared to determine the reliability coefficient.

The alternative forms method with different assessment instruments accounts for different instruments' content as a source of error. Similar approaches as described above for alternative forms with different assessment teams could be followed.

The split-halves method can be applied by having a number of trials assessments use an assessment instrument that is somewhat longer than usual. This instrument would subsequently be divided into two halves and the reliability coefficient for each half and the total instrument computed.

One difficulty with the split-halves method, however, is that the reliability estimate depends on the way the instrument is divided into two halves. For example, for a 10 question instrument, there are 126 possible different splits [9], and hence 126 different split-halves reliability estimates. The most common procedure is to take even numbered items on an instrument as one part and the odd numbered ones as the second part.

An internal consistency method can be applied by computing the Cronbach alpha coefficient from the results of several SPICE-conformant assessments. To allow internal consistency methods to be used, these assessments would at least have to be using the same instrument (or overlapping instruments) and follow the same assessment process. In the context of the SPICE trials, however, different participating organizations are expected to have different priorities, and would therefore prefer to be assessed against different dimensions of capability (e.g., engineering processes vs. support processes). This means that using the same instrument or overlapping instruments for a sufficiently large number of trial assessments may be difficult.

If one were to adopt the generalizability theory framework, then multiple sources of error could be investigated simultaneously. The approaches that would be utilized under the generalizability theory framework are similar to those described above for the alternative forms method. The difference would be in explicitly accounting for *multiple* sources of error in the same study.

The test-retest and the delayed alternative forms methods account for different occasions as a source of error. In the context of SPICE, there are at least three important difficulties in conducting studies that account for different occasions as a source of error.

The first difficulty is the expense of conducting assessments at more than one point in time. Given that prior experience has identified the costs of process assessments as a concern [8][39], the costs of repeated assessments would be perceived as substantial. It is already difficult enough to find sponsorship for a single assessment.

Second, it is not obvious that a low reliability coefficient obtained using a test-retest or delayed alternative forms method indicates low reliability. For example, a likely explanation for a low coefficient is that the organization's software process capability has changed between the two assessment occasions. For instance, the initial assessment and its results might sensitize an organization to specific weaknesses and prompt them to initiate an improvement effort that influences the results of subsequent assessments.

Third, carry-over effects between assessments may lead to an over-estimate of reliability. For instance, the reliability coefficient can be artificially inflated due to memory effects. Examples of memory effects are the assessees knowing the 'right' answers that they have learned from the previous assessments and, assessors remembering responses from previous assessments, and, deliberately or otherwise, repeating them in an attempt to maintain the consistency of results.

3 Validity

3.1 The Importance of Validity

Validity of measurement is defined as the extent to which a measurement procedure is measuring what it is purporting to measure [40]. During the process of validating a measurement procedure one attempts to collect evidence to support the types of inferences that are to be drawn from measurement scores [15]. In the context of SPICE, concern with validity is epitomized by the question: *"are SPICE-conformant assessments really measuring best software process practices?"*

Validity is related to reliability in the sense that reliability is a necessary but insufficient condition for validity. The differences between reliability and validity are illustrated below by way of two examples.

For example, assume one seeks to measure intelligence by having children throw stones as far as they could. The distance the stones are thrown on one occasion might correlate highly with how far they are thrown on another occasion. Thus, being repeatable, the stone-throwing measurement procedure would be highly reliable. However, the distance that stones are thrown would not be considered by most observers to be a valid measure of intelligence.

As another example, consider a car's fuel gauge that systematically shows ten liters higher than the actual level of fuel in the gas tank. If repeated readings of fuel level are taken under the same conditions, the gauge will yield consistent (and hence reliable) measurements. However, the gauge does not give a valid measure of fuel level in the gas tank.

Investigating the validity of SPICE conformant assessments during the SPICE trials is an important objective. This importance becomes clear when one considers that the empirical evidence supporting the efficacy of many of the practices codified in SPICE is far from overwhelming or even convincing on scientific grounds[5].

For SPICE, there are three types of validity that are of interest. These are: content validity, criterion-related validity, and construct validity. There is a greater concern with criterion-related validity in the software process community, so greater emphasis will be placed on it in the ensuing discussion.

3.2 Content Validity

Content validity is defined as the representativeness or sampling adequacy of the content of a measuring instrument [40]. Ensuring content validity depends largely on expert judgement.

In the context of SPICE, expert judgement would ensure that SPICE-conformant measurement procedures are at least perceived to measure best software process practices. In order to explain content validation for SPICE, it is necessary to first briefly overview one of the core SPICE documents[6], the Baseline Practices Guide (BPG).

The purpose of the BPG is to *"document the set of practices essential to good management of software engineering"* [59]. The practices in the BPG are categorized into either one of two groups. The first group is the *Base Practices*. The second group is the *Generic Practices*.

A base practice is defined as *"a software engineering or management activity that addresses the purpose of a particular process, and thus belongs to it. Consistently performing the base practices associated with a process, and improving how they perform, will help in consistently achieving its purpose"* [59]. An example of a process is *Develop System Requirements and Design*. Base practices that belong to this process include: *Specify System Requirements*, *Describe System Architecture*, and *Determine Release Strategy*.

A generic practice is defined as *"an implementation or institutionalization practice (activity) that enhances the capability to perform a process"* [59]. Generic practices are grouped into Common Features. An example of a Common Feature is *Disciplined Performance*. A generic practice that belongs to this Common Feature stipulates that data on the performance of the process must be recorded.

The BPG defines the content domain of best software process practices through the Base and Generic Practices. It is hypothesized that this content domain is applicable across software organizations of different sizes, in different industrial sectors, and that follow different software

5 The lack of empirical studies, and hence empirical evidence, is not unique to SPICE, but is a general characteristic of software engineering [6] and computer science [46].

6 The names of some of the SPICE documents may be changed before the completion of standardization. The names referred to here are those that are currently used.

development life cycles. The BPG has been reviewed by experts from industry and academe to ensure that it adequately covers the content domain, and it has been revised accordingly. Furthermore, coverage of the BPG is also being evaluated during actual trial applications of SPICE. The trial applications are conducted by selected experienced assessors who will provide the coverage feedback.

All SPICE-conformant assessments must be based on a set of practices that at a minimum include those defined in the BPG or in a conformant variant of the BPG for the processes assessed. Given the extensive 'arm-chair' based and real-application based reviews of the BPG, one would be confident that the BPG provides reasonable coverage of the "best software process practices" domain.

To further ensure content validity, it is necessary that all assessment instruments include questions that adequately sample from the content domain. Another SPICE document, the Assessment Instrument, prescribes guidelines for creating SPICE-conformant assessment instruments. Furthermore, an exemplar assessment instrument is scheduled for development as part of the SPICE project. Both of these will undergo the same validation procedure as the BPG to ensure that SPICE-conformant assessment instruments have a high level of content validity.

3.3 Criterion-Related Validity

With criterion-related validity one attempts to determine the magnitude of relationship (using a correlation coefficient or some other measure of association) between the score an organization obtains in a SPICE-conformant assessment and some other criterion. Two criteria of interest to SPICE are: (a) performance measures (for example, number of software defects post-release, productivity, cost per line of code, user satisfaction etc.), and (b) other measures of software process capability (for example, those based on the SEI's CMM or Bootstrap). These two criteria differentiate between two types of criterion-related validity respectively: (a) predictive validity, and (b) concurrent validity.

3.3.1 Predictive Validity

Perhaps the most important type of validity of concern to the software process community is predictive validity. For instance, in the context of CMM-based assessments, Hersh [36] states *"despite our own firm belief in process improvement and our intuitive expectation that substantial returns will result from moving up the SEI scale - we still can't prove it"* (although, there is now some initial evidence [33]). Also, Fenton [27] notes that evaluating the validity of the SEI's process maturity scheme is a key contemporary research issue.

There have been some recent efforts at empirically investigating the predictive validity of software process assessments, including those based on the CMM, such as [33][35]. The SPICE trials are attempting to build upon this previous work. However, we face serious methodological difficulties and constraints on available resources.

Performance measures used in validity studies can be objective or subjective[7]. An example of an objective measure is lines of code per hour. An example of a subjective measure is user satisfaction.

Ideally, performance measures would be gathered consistently across all organizations involved in a validity study. All measures would be defined in the same way (e.g., line of code counting procedures), and measurement instruments would be sufficiently generic to be applicable to all organizations and their businesses. While we think it is tractable, this requirement will present considerable difficulty in the SPICE trials, especially since the trials are being conducted globally.

If a strong relationship is found between SPICE-conformant assessment scores and performance measures, then this would provide strong *initial* evidence supporting SPICE validity. However, if weak or no relationships are identified, then we have an interpretation problem. Finding no empirical relationship can be interpreted in at least three ways: (a) the empirical study was flawed, (b) the hypothesized relationship between the assessment score and the performance measure is wrong, and/or (c) SPICE-conformant assessments do not measure best software process practices. Thus, if no relationships are identified, then great caution should be taken in drawing conclusions from a predictive validity study.

Another problem with predictive validity studies is identifying appropriate measures of performance. Two types of measures can be discerned: (a) project effectiveness measures, and (b) organizational

7 The classification of metrics as either objective or subjective is common in software engineering, for example see [7]. This classification is therefore used here.

effectiveness measures. Project effectiveness measures evaluate the outcomes of a single project or a phase of a single project. Organizational effectiveness measures evaluate the performance of a whole software organization. Possible measures of both types are presented below.

A recent review by Krasner [43] of the payoff for software process improvement identifies a number of possible project effectiveness measures. These include productivity, cost of rework, defect density, early defect detection rate, number of defects discovered by the customer, cost per line of code, and predictability of costs and schedules. Another study conducted by the SEI [35] utilized similar measures. Rozum [57] also reviews a number of possible measures including mean time between failures and availability.

A difficulty with attempting to collect performance data is that projects that have low scores on SPICE-conformant assessments generally will not collect or maintain such data, and hence they would have to be excluded from a validity study. This would reduce the variation in the performance measure, and thus reduce (artificially) the validity coefficients.

An alternative approach is to collect subjective data at the time a validity study is conducted. A recent study investigating the relationship between software process and project performance [19] utilized Likert-type scales to measure perceived software quality and perceptions of meeting (schedule and budget) targets. A survey investigating the relationship between process maturity and performance utilized Likert-type scales to measure product quality, productivity, meeting schedule and budget targets, staff morale, and perceptions of customer satisfaction [33]. Another study investigating the relationship between organizational maturity and requirements engineering success [26] utilized a requirements engineering success instrument [25]. Davis [18] has developed an instrument to measure the perceived usefulness and ease of use of software. Rozum [57] considers customer satisfaction, and Krishnan [44] in his study of the relationship between software product and service characteristics and customer satisfaction, measured customer satisfaction using a five-point ordinal scale. Where the domain of analysis is business information systems, commonly used measures of Information Systems success include system usage (empirical studies where system usage was measured include [56][60]), and user satisfaction [20].

An obvious approach for measuring organizational effectiveness is to aggregate project effectiveness measures for a representative sample of projects in a software organization. Alternatively, where the domain of analysis is business information systems, a number of possible organizational effectiveness measures can be used. For example, Mahmood and Soon [47] have defined and operationalized a set of strategic variables that are potentially affected by Information Technology (IT), and can be used to evaluate the impact of IT on an entire organization. More commonly used measures are those of overall use of Information Systems (e.g., see [5]) and of user information satisfaction with the Information Systems function [4][38]. However, measures of organizational effectiveness are relatively controversial. Miller [49] notes that: *"It appears futile to search for a precise measure or set of measures of [Information Systems] effectiveness that will be common across all organizations. Criteria for effectiveness in a single organization can be expected to vary with changing value structures, levels in the organization and phases in organizational growth"*.

3.3.2 Concurrent Validity

Concurrent validity is of interest if we want to answer the question: *"are scores obtained from SPICE-conformant assessments related to the scores obtained using other software process assessment methods?"* According to common expectations, SPICE scores should be highly related to the scores on other assessment methods.

This expectation could be tested by first identifying organizations that have just recently undergone, say, a Bootstrap based assessment. Subsequently they would undergo a SPICE-conformant assessment. The correlation coefficient (or some other measure of association) would be computed between the two scores. If the magnitude of the association is found to be high, then this provides evidence that SPICE-conformant assessments are measuring the same thing as the other method(s).

If the association is high and if one makes the assumption that the other non-SPICE-related methods are measuring best software process practices, then one can have strong evidence to the validity of SPICE. However, given that there is little scientific evidence to that effect (i.e., that the other methods are indeed measuring best software process practices), this approach to validation would not be too convincing.

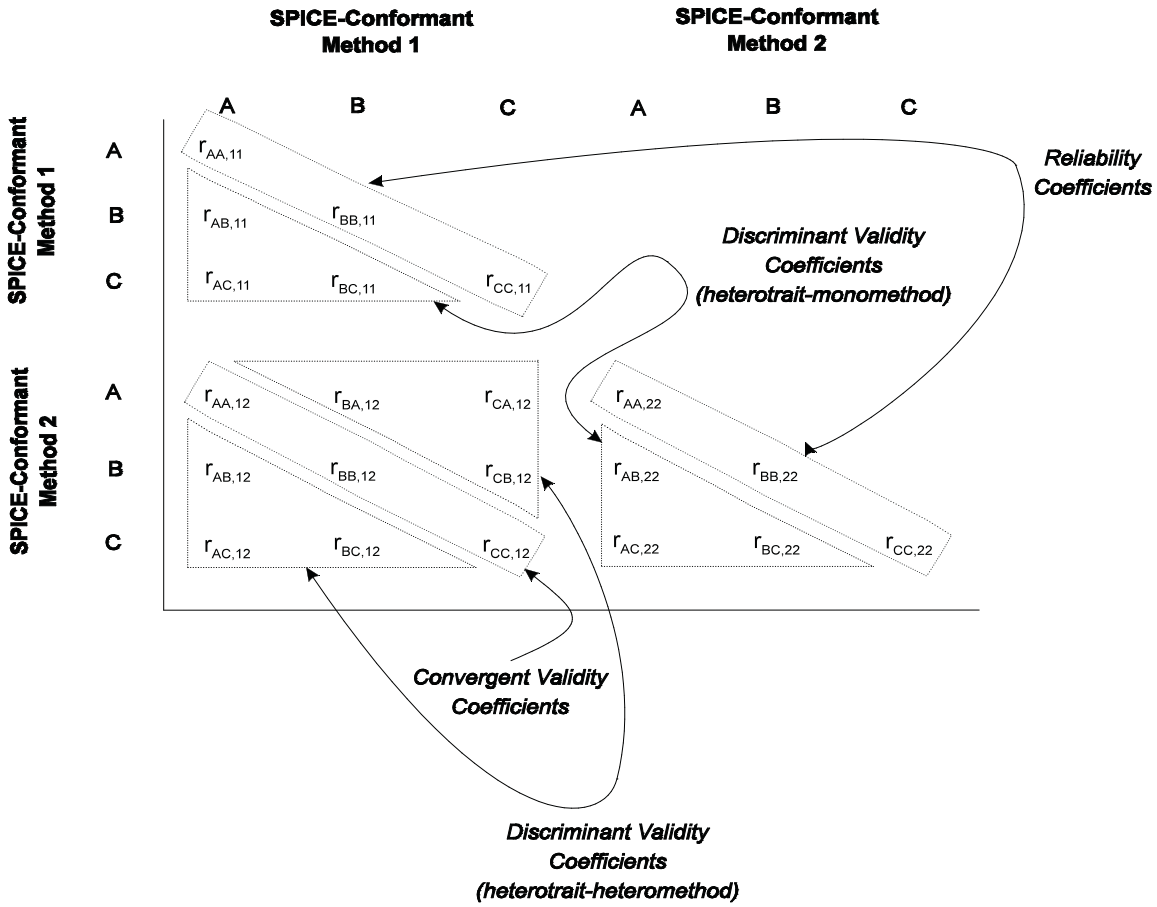

Figure 5: The MultiTrait-MultiMethod matrix approach for evaluating construct validity.

Furthermore, a concurrent validity study would face the same difficulties as a test-retest study. These difficulties were mentioned in section 2 and they are: high expense, interpreting low validity coefficients, and carry-over effects.

3.4 Construct Validity

Construct validity[8] is an operational concept that asks whether two or more SPICE-conformant assessments measure the same concept (best software process practices), and whether they can differentiate between the *different dimensions* of that concept (e.g., best engineering practices, best customer supplier practices etc.). The above distinctions are often referred to as *convergent validity* and *discriminant validity* respectively [11].

8 The exact definition of construct validity and the procedures for construct validation tend to vary across the literature. For instance, Bagozzi [3] considers providing evidence of what we have defined here as reliability to be part of construct validation. The way we have defined and grouped concepts and methods in this paper, however, is the one more commonly found in the literature.

Convergent and discriminant validities can be evaluated using the MultiTrait-MultiMethod (MTMM) matrix [11]. The basic idea behind the MTMM approach is to assess several organizations on two or more dimensions using two or more different methods. The obtained data are analyzed in a matrix.

An MTMM matrix is shown in Figure 5. In this matrix there are two measurement methods (e.g., two SPICE-conformant assessment methods), and three dimensions (e.g., A: best engineering practices, B: best customer supplier practices, and C: best support practices).

An MTMM matrix has three types of coefficients: reliability coefficients, convergent validity coefficients, and discriminant validity coefficients. Reliability coefficients have been discussed earlier in this paper.

Convergent validity coefficients are correlations between measures of the same dimension using different measurement methods. Evidence of high convergent validity exists when these correlations

are high. Convergent validity indicates that the scores are less likely to be artifacts of the chosen measurement procedure.

Discriminant validity coefficients can either be the correlations between measures of the different dimensions using the same method of measurement (these are referred to as heterotrait-monomethod coefficients), or correlations between different dimensions using different methods of measurement (these are referred to as heterotrait-heteromethod coefficients). Evidence of high discriminant validity exists when these correlations are lower than the reliability and convergent validity coefficients. Discriminant validity indicates that methods of measurement can differentiate between the different dimensions.

In the context of SPICE, an MTMM approach would necessitate multiple assessments of the same organization. As mentioned earlier, it is already difficult to find sponsorship for a single assessment. Furthermore, several organizations would have to be assessed on the same dimensions. This may be difficult since the priorities of participating organizations are likely to differ.

Another approach for evaluating convergent and discriminant validities is through factor analysis [51]. Factor analysis is a multivariate technique for 'clustering' variables. These variables would be the questions on multiple dimensions. If the emerging 'clusters' match the dimensions, then this is evidence of construct validity. The logic behind using factor analysis is that variation among a number of questions that form a cluster can be attributed to variation among organizations on one common underlying factor (e.g, best customer-supplier practices).

4 Discussion

While we have presented some realistic approaches for conducting reliability and validity studies, it must be noted that there exist limitations on the SPICE trials, and that some tradeoffs are necessary in the design and conduct of the studies. Limitations and tradeoffs are not unique to SPICE, but are of equal concern to any scientist conducting empirical research in software engineering.

4.1 Limitations

Participation in the SPICE trials by experienced assessors and by organizations is on a voluntary basis. Although those participating are expected to gain many benefits from their efforts, it would not be prudent during the trials planning to assume a sufficiently large number of participants. Thus, it is assumed that sample sizes will probably be small, at least for the initial phases of the SPICE trials.

When reliability and validity coefficients are estimated from small samples, sampling errors are relatively large. This means that the statistical power[9] of the inferential procedures used to analyze the data from the studies is likely to be substantially reduced.

One approach to increasing the statistical power in such a situation is to use parametric as opposed to non-parametric tests, since these are, in general, more powerful [42]. However, parametric tests assume specific distributions in the data. Given the nature of software process assessment data, it is expected that they will not be 'well-distributed'. Cohen [14], however, suggests the use of non-parametric tests only under *extreme* violations of the assumptions of the parametric tests. Some of these assumptions can be tested once more SPICE trials data are collected.

4.2 Tradeoffs

As is common in many empirical research studies, a necessary tradeoff exists in the selection of a particular empirical research strategy (e.g., field experiments vs. surveys vs. laboratory experiments etc.). McGrath [48] makes the point clearly: *"all research strategies are 'bad' (in the sense of having serious methodological limitations); none of them are 'good' (in the sense of being even relatively unflawed). So, methodological discussions should not waste time arguing about which is the right strategy, or the best one; they are all poor in an absolute sense"*.

One possible approach for alleviating such concerns is to follow a multimethod empirical research strategy. The logic of the multimethod strategy is [10] *"to attack a research problem with an arsenal of methods that have nonoverlapping weaknesses in addition to their complementary strengths"*. This strategy effectively addresses monomethod bias in the results of a study. A multimethod strategy is being considered as part of the SPICE trials planning. Although, that strategy is constrained by limited resources.

9 Statistical power is defined as the probability that a statistical test will correctly reject the null hypothesis.

Rarely does a single research project satisfy a complete multimethod strategy, but the collective results of an emerging discipline *can* use multiple methods. This is an important point that requires strong emphasis. Software engineering researchers should conduct their own empirical studies, using different methods, on the reliability and validity of SPICE-conformant assessments subsequent to the SPICE trials. It is through such external studies that one can gain greater confidence in the results of the SPICE trials. Furthermore, a diversity of empirical research methods that are used in a scientific discipline is itself a sign of the discipline's own maturity [13].

5 Conclusions

In this paper we have presented some important theoretical and practical considerations pertinent to the trials phase of the SPICE project. These considerations are driven by two of the objectives of the trials: to determine (a) the reliability, and (b) the validity of SPICE-conformant assessments. Of course, these considerations and the questions they raise are equally relevant to other software process assessment methods in existence today.

For the reliability objective, we presented two theoretical frameworks for estimating reliability: (a) the classical test theory framework, and (b) the generalizability theory framework. For the validity objective, we presented three types of validation approaches: (a) content validation, (b) criterion-related validation, and (c) construct validation. In addition, we discussed reliability estimation and validation methods and their applicability in the context of the SPICE trials.

We have also attempted to paint a realistic picture of the constraints and limitations of the SPICE trials. These constraints and limitations are important because they will shape what eventually emerges from the SPICE trials, and the degree of confidence one can place in their results.

The SPICE trials will *not provide "conclusive proof "* about the extent of reliability and validity of SPICE-conformant assessments. The trials *can* provide some initial evidence based on well designed and executed applied research. The burden is upon the software engineering research community to replicate and extend the trials' studies, and to support or disconfirm their findings.

Acknowledgements

The authors wish to thank Jerome Pesant and members of the Software Engineering Laboratory at McGill University for their comments on an earlier version of this paper.

References

[1] D. Adams, R. Nelson, and P. Todd: "Perceived usefulness, ease of use, and usage of information technology: A replication". In *MIS Quarterly*, pages 227-247, June 1992.

[2] K. Amoako-Gyampah and K. White: "User involvement and user satisfaction: An exploratory contingency model". In *Information and Management*, 25:1-10, 1993.

[3] R. Bagozzi: *Causal Models in Marketing*, John Wiley, 1980.

[4] J. Bailey and S. Pearson: "Development of a tool for measuring and analyzing computer user satisfaction". In *Management Science*, 29(5):530-545, May 1983.

[5] J. Baroudi, M. Olson, and B. Ives: "An empirical study of the impact of user involvement on system usage and information satisfaction". In *Communications of the ACM*, 29(3):232-238, March 1986.

[6] V. Basili: "The experimental paradigm in software engineering". In *Experimental Software Engineering Issues: Critical Assessment and Future Directions*, H. D. Rombach, V. Basili, and R. Selby (eds.), Springer-Verlag, 1993.

[7] V. Basili and D. Rombach: "The TAME Project: Towards Improvement-Oriented Software Environments". In *IEEE Transactions on Software Engineering*, 14(6):758-773, June 1988.

[8] J. Besselman, P. Byrnes, C. Lin, M. Paulk, and R. Puranik: "Software Capability Evaluations: Experiences from the field". In *SEI Technical Review*, 1993.

[9] G. Bohrnstedt: "Reliability and validity assessment in attitude measurement". In *Attitude Measurement*, G. Summers (ed.), Rand McNally and Company, 1970.

[10] J. Brewer and A. Hunter: *Multimethod Research: A Synthesis of Styles*, Sage Publications, 1989.

[11] D. Campbell and D. Fiske: "Convergent and discriminant validation by the multitrait-multimethod matrix". In *Psychological Bulletin*, pages 81-105, March 1959.

[12] D. Card: "Capability evaluations rated highly variable". In *IEEE Software*, pages 105-106, September 1992.

[13] M. Cheon, V. Grover, and R. Sabherwal: "The evolution of empirical research in IS: A study in IS maturity". In *Information and Management*, 24:107-119, 1993.

[14] J. Cohen: "Some statistical issues in psychological research". In *Handbook of Clinical Psychology*, B. Woleman (ed.), McGraw-Hill, 1965.

[15] L. Cronbach: "Test validation". In *Educational Measurement*, R. Thorndike (ed.), American Council on Education, 1971.

[16] L. Cronbach: “Coefficient alpha and the internal consistency of tests”. In *Psychometrika*, pages 297-334, September 1951.
[17] L. Cronbach, G. Gleser, H. Nanda, and N. Rajaratnam: *The Dependability of Behavioral Measurements: Theory of Generalizability for Scores and Profiles*, John Wiley, 1972.
[18] F. Davis: “Perceived usefulness, perceived ease of use, and user acceptance of information technology”. In *MIS Quarterly*, pages 319-340, September 1989.
[19] C. Deephouse, D. Goldenson, M. Kellner, and T. Mukhopadhyay: “The effects of software processes on meeting targets and quality”. In *Proceedings of the Hawaiian International Conference on Systems Sciences*, vol. 4, pages 710-719, January 1995.
[20] W. Doll and G. Torkzadeh: “The measurement of end-user computing satisfaction”. In *MIS Quarterly*, pages 259-274, June 1988.
[21] A. Dorling: “SPICE: Software Process Improvement and Capability dEtermination”. In *Information and Software Technology*, 35(6/7):404-406, June/July 1993.
[22] J-N Drouin: “Software quality - An international concern”. In *Software Process, Quality & ISO 9000*, 3(8):1-4, August 1994.
[23] J-N Drouin: “The SPICE project: An overview”. In *Software Process Newsletter*, IEEE Computer Society, No. 2, pages 8-9, Winter 1995.
[24] G. Dunn: *Design and Analysis of Reliability Studies: The Statistical Evaluation of Measurement Errors*, Oxford University Press, 1989.
[25] K. El Emam and N. H. Madhavji: “Measuring the success of requirements engineering processes”. In *Proceedings of the Second IEEE International Symposium on Requirements Engineering*, pages 204-211, 1995.
[26] K. El Emam and N. H. Madhavji: “The reliability of measuring organizational maturity”. Submitted for publication, 1995.
[27] N. Fenton: “Objectives and context of measurement/experimentation”. In *Experimental Software Engineering Issues: Critical Assessment and Future Directions*, H. D. Rombach, V. Basili, and R. Selby (eds.), Springer-Verlag, 1993.
[28] N. Fenton, B. Littlewood, and S. Page: “Evaluating software engineering standards and methods”. In *Software Engineering: A European Perspective*, R. Thayer and A. McGettrick (eds.), IEEE Computer Society Press, pages 463-470, 1993.
[29] N. Fenton and S. Page: “Towards the evaluation of software engineering standards”. In *Proceedings of the Software Engineering Standards Symposium*, pages 100-107, 1993.
[30] N. Fenton, S-L Pfleeger, S. Page, and J. Thornton: “The SMARTIE standards evaluation methodology”. *Technical Report* (available from the Centre for Software Reliability, City University), 1994.
[31] T. Frick and M. Semmel: “Observer agreement and reliabilities of classroom observational measures”. In *Review of Educational Research*, 48:157-184, 1978.
[32] D. Galletta and A. Lederer: “Some cautions on the measurement of user information satisfaction”. In *Decision Sciences*, 20:419-438, 1989.
[33] D. Goldenson and J. Herbsleb: “What happens after the appraisal? A survey of process improvement efforts”. Paper presented at the *1995 SEPG Conference*, 1995.
[34] J. Guilford: *Psychometric Methods*, McGraw-Hill, 1954.
[35] J. Herbsleb, A. Carleton, J. Rozum, J. Siegel, and D. Zubrow: “Benefits of CMM-Based Software Process Improvement: Initial Results”. *Technical Report*, CMU/SEI-94-TR-13, Software Engineering Institute, August 1994.
[36] A. Hersh: “Where's the Return on Process Improvement?”. In *IEEE Software*, page 12, July 1993.
[37] W. Humphrey and B. Curtis: “Comments on ‘A Critical Look’”. In *IEEE Software*, pages 42-46, July 1991.
[38] B. Ives, M. Olson, and J. Baroudi: “The measurement of user information satisfaction”. In *Communications of the ACM*, 26(10):785-793, 1983.
[39] Japan SC7 WG10 SPICE Committee: “Report of Japanese trial process assessment by SPICE method”. *A SPICE Project Report*, 1994.
[40] F. Kerlinger: *Foundations of Behavioral Research*, Holt, Rinehart, and WInston, 1986.
[41] M. Konrad: “On the horizon: An international standard for software process improvement”. In *Software Process Improvement Forum*, pages 6-8, September/October 1994.
[42] H. Kraemer and S. Thiemann: *How Many Subjects? Statistical Power Analysis in Research*, Sage Publications, 1987.
[43] H. Krasner: “The Payoff for Software Process Improvement (SPI): What it is and how to get it”. In *Software Process Newsletter*, IEEE Computer Society, No. 1, pages 3-8, September 1994.
[44] M. Krishnan: “Software product and service design for customer satisfaction: An empirical analysis”. *Technical Report*, Graduate School of Industrial Administration, Carnegie Mellon University, 1993.
[45] F. Lord and M. Novick: *Statistical Theories of Mental Test Scores*, Addison-Wesley, 1968.
[46] P. Lukowicz, E. Heinz, L. Prechelt, and W. Tichy: “Experimental evaluation in computer science: A quantitative study”. *Technical Report*, 17/94, Department of Informatics, University of Karlsruhe, 1994.
[47] M. Mahmood and S. Soon: “A comprehensive model for measuring the potential impact of information technology on organizational strategic variables”. In *Decision Sciences*, 22(4):869-897, 1991.
[48] J. McGrath: “Dilemmatics: The study of research choices and dilemmas”. In *Judgement Calls in Research*, J. McGrath, J. Martin, and R. Kulka (eds.), Sage Publications, 1982.
[49] J. Miller: “Information systems effectiveness: The fit between business needs and system capabilities”. In *Proceedings of the 10th International Conference on Information Systems*, pages 273-288, 1989.
[50] J. Neter, W. Wasserman, and M. Kutner: *Applied Linear Statistical Models: Regression, Analysis of Variance, and Experimental Designs*, Irwin, 1990.
[51] J. Nunnally: *Psychometric Theory*, McGraw-Hill, 1978.

[52] M. Paulk and M. Konrad: “Measuring process capability versus organizational process maturity”. In *Proceedings of the 4th International Conference on Software Quality*, 1994.
[53] M. Paulk and M. Konrad: “ISO seeks to harmonize numerous global efforts in software process management”. In *Computer*, pages 68-70, April 1994.
[54] S-L Pfleeger: “The language of case studies and formal experiments”. In *Software Engineering Notes*, pages 16-20, October 1994.
[55] S-L Pfleeger, N. Fenton and S. Page: “Evaluating software engineering standards”. In *Computer*, pages 71-79, September 1994.
[56] D. Robey: “User attitudes and management information system use”. In *Academy of Management Journal*, 22(3):527-538, September 1979.
[57] J. Rozum: “Concepts on measuring the benefits of software process improvements”. *Technical Report*, CMU/SEI-93-TR-009, Software Engineering Institute, 1993.
[58] V. Sethi and W. King: “Construct measurement in information systems research: An illustration in strategic systems”. In *Decision Sciences*, 22:455-472, 1991.
[59] The SPICE Project: *Baseline Practices Guide* (draft), 1994.
[60] A. Srinivasan: “Alternative measures of system effectiveness: Associations and implications”. In *MIS Quarterly*, 9(3):243-253, September 1985.
[61] A. Subramanian and S. Nilakanta: “Measurement: A blueprint for theory-building in MIS”. In *Information and Management*, 26:13-20, 1994.
[62] P. Tait and I. Vessey: “The effect of user involvement on system success: A contingency approach”. In *MIS Quarterly*, pages 91-108, March 1988.
[63] G. Torkzadeh and W. Doll: “The test-retest reliability of user involvement instruments”. In *Information and Management*, 26:21-31, 1994.

Appendix

This appendix provides some general guidelines for increasing the reliability of software process assessment methods. These guidelines are intended for those developing and/or using assessment methods and/or frameworks.

1 Standardize Assessment Procedures
The procedures used for an assessment must be standardized and individual assessments must follow them closely to ensure consistency. In the case of assessment instruments, instructions concerning the purpose and how to determine responses and judge scores should be provided. In the case of interviews, the conduct of the interviews (e.g., assurance of confidentiality and the type and scope of evidence inspected) should be defined.

2 Training of Assessors
Assessors should be trained in the assessment procedure and should have extensive experience with software development and maintenance. Furthermore, there should be a consistency in the qualifications of the assessors following a particular assessment procedure.

3 Increasing the Length of the Assessment Instrument
Reliability estimates utilize the assessment scores. The more questions asked about the capability of an organization, the more likely that the reliability estimates are increased. Of course, if the added questions have nothing to do with maturity, then increasing the length of the instrument may reduce reliability. However, it is assumed that added questions are chosen as carefully as the original questions and that they will not reduce the average inter-item correlations.

4 Defined Sampling Criteria
In assessment procedures where a sample of an organization's projects are assessed, and these are used as an indicator of overall organizational capability, specific sampling criteria should be specified. These sampling criteria should be applied consistently in all assessments claiming to follow a particular assessment procedure.

5 Using Multiple Point Scales
Determining the number of points on a scale involves a tradeoff between losing some of the discriminative powers of which the assessors are capable (with too few points) and having a scale that is too fine and hence beyond the assessors' powers of discrimination (with too many points). In general, it has been found that there is an increase in reliability as the number of points increases from 2 to 7, after which it tends to level off [34][51].

6 Having a Validation Cycle
Such a cycle involves validating the information that the assessors have initially gathered. This may involve corroborative interviews and (further) inspections of documents. This would seem to be more important were the assessors are external to the assessed organization and when there is a danger of misrepresentation by the assessees.