The Measurement of Consumer Attitudes Concerning Software Quality

Michael W. Usrey
University of Colorado
Engineering Management Program
Boulder, CO 80309-0435
michael.usrey@colorado.edu

Kevin J. Dooley
Arizona State University
Dept. of Industrial & Management Systems Engineering
Tempe, AZ 85287-5906
kevin.dooley@asu.edu

Cite: Usrey, M., and K. Dooley (1998), "The Measurement of Consumer Attitudes Concerning Software Quality," *Quality Management Journal*, 5(2): 42-57.

Bios

Michael Usrey is an Assistant Professor at University of Colorado, Boulder. He teaches in the Lockheed-Martin Engineering Management Graduate Program there, and has research interests in software engineering, engineering management, and industrial engineering. He obtained his Ph.D. in Industrial Engineering from the University of Minnesota. Prof. Usrey is President and co-founder of Protocol Communications, an Internet service provider. He has previously worked for Honeywell Aviation Systems and Seagate Technologies.

Kevin Dooley is a Professor at Arizona State University. He has a joint appointment in the Departments of Management, and Industrial and Management Systems Engineering. He previously taught at University of Minnesota, and has a PhD in Mechanical Engineering from the University of Illinois. He teaches and does research in the areas of quality engineering, quality management, new product development, organizational change, and complex systems. Prof. Dooley is on the editorial board of *Quality Management Journal, Journal of Operations Management,* and *Nonlinear Dynamics, Psychology, & the Life Sciences.*

ABSTRACT

This paper is concerned with the development of an instrument aimed at the measurement of consumer attitudes concerning software quality. The operationalization of the construct software quality includes the dimensions of communication, completeness, features, flexibility, independence, simplicity, stability, and maturity. Ordinal scale development methods are used to develop objective, operational measures of these dimensions. An on-line, Internet-based survey tool was developed to facilitate the large number of responses required for survey validation. The instrument will help solve important problems of software development productivity, software engineering research and survey research. Results show that relevant software consumer attitudes can be consistently and reliably observed and quantified. Results also confirm earlier studies that report benefits from the use of computer-mediated communication for survey research. Demographic data reveals an unbiased survey and provides confirmation of several trends existing today in the software marketplace.

.

key words: software, quality measurement, survey, Guttman scales

1 INTRODUCTION

Software is pervading, directly and indirectly, on almost every aspect of our lives [Bucci, 1995]. According to Long [1984], "There is no escape from the effects of computerization." Computers are a driving force in many areas of society [Hicks, 1984]. The evolution and spread of information system technology has provided a number of firms with the means to become more effective competitors [Johnson, 1989]. A host of economic and other factors make this increasing impact inevitable [Goldberg and Lorin, 1982]. With personal computers and embedded systems, software pervades almost every aspect of our lives [Schach, 1990]. The need to understand, control, and design quality software is unprecedented [Gibbs, 1994].

Numerous works have focused on how to provide quality in the development process [Albrecht, 1979; Boehm, 1984; Cho, 1987; Curtis et al., 1979; Department of Defense, 1985; Dreger, 1989; Fagan, 1976; Humphrey and Sweet, 1987; Jones, 1990; Kemerer, 1987; Schulmeyer, 1987]. Recent work [ISO/IEC, 1991; Usrey and Dooley, 1996] has also focused on software product quality. Our work is motivated by research done by Boehm et al. [1978] which identified characteristics of software quality and by the work of Garvin [1988] which outlined the dimensions of product quality. Our analysis is driven by customer perceptions of quality, and thus defines and operationalizes software quality from the perspective of customer satisfaction with a software product.

With computers on our desks and embedded in almost every technology we use, software pervades almost every aspect of our lives [Schach, 1990]. Software has become the central component in many complex activities; the challenge of producing it requires specialized and powerful techniques [Jones, 1990]. As society becomes more dependent on software, there is a growing demand for improving its quality [Bucci, 1995]. Unfortunately, there is no practical way to show that one program is better than another [Cho 1987]. This is a compelling reason to evolve software quality assessment from a craftsman activity towards a well-organized process. The measurement of software quality allows one to understand cost-benefit issues in software design and quality testing [Dobbins and Buck, 1987].

Field quality measurements are a key activity during the software maintenance phase [ISO 9000-3, 1991; ISO 9001, 1994; ISO 9004-1, 1994; ISO 9004-4, 1994, Paulk, 1993; Coletta, 1995; as cited by Cicu, 1995]. Questionnaires are an established means of measuring user satisfaction during this phase [Corbett, Porteous and Kirakowski, 1992]. Customer surveys are an important tool for assessing customer satisfaction during the operational phase of the product life cycle [Jones, 1990]. The authors believe that the survey developed in the current research would be widely useful not only for this type of activity, but also for feedback in a rapid-prototyping paradigm for software development. Reuse of software and related artifacts improves quality, productivity and cost [Meyer, 1988; Mullin, 1989; Karlsson and Morel, 1995]. The scales in the research presented here are intended to become such reusable artifacts.

A major goal of the research to be presented is to use survey techniques to add operational measures to nominal definitions of quality dimensions. Quality assessment is closely related to attitudes [Guaspari, 1985; Prell and Sheng, 1984; Ramamoorthy et al., 1984], so attitude research methods were applied. Methods of operationalizing attitude concepts include indexes and scales [Babbie, 1979]. Our structure is based on behaviorally anchored rating scales (BARS), which combine aspects of both indexes and scales [Scott, 1985]. Anastasi [1988] describes five steps in attitude testing: formulation of trait or construct nominal definitions, prepare and administer test items, item analysis, demographic analysis, and validation against external criteria. Anastasi's methodology was used to construct and test our research instrument.

2 INSTRUMENT DESIGN

A summary of the instrument design follows. First, previous research efforts were drawn upon for the trait nominal definitions. Then, heuristic techniques were used to design questionnaire items that are likely to result in unidimensional scales and on-line survey techniques were employed to administer the survey. Third, measures of dimensionality and association were calculated to empirically evaluate the unidimensionality and sensitivity of each scale. Scales were purified to leave only discriminating scale items during the third phase. Demographic outlier analysis was performed. In the fifth and final phase, the purified scale was validated with a control group and against external criteria.

2.1 Content and Principle Component Analysis

As previously stated, the first step in attitude testing is nominally defining the attitude constructs under study or latent variables. Source materials for identifying such nominal definitions for software product quality include the work of McCall [1977], Boehm et al [1978], Garvin [1988], Zeithaml, Parasuramna, and Berry [1990], Misterek, Anderson and Dooley [1990], and ISO/IEC [1991]. Taken together, these sources produce a large number of software quality constructs. There are cases of overlaps and inconsistencies between latent variables. Terms and nominal definitions are sometimes overloaded. For example, the concept of a product meeting its stated requirements is referred to by ISO as "functionality," by Garvin as "conformance," and by Boehm et al. as "completeness." Previous efforts by the researchers [Usrey and Dooley, 1996] sought to integrate and validate these latent variables and assess their relative contribution to software quality.

Validation of attitude constructs was accomplished using the technique of manifest content analysis [ibid]. Manifest content analysis is a retrospective analysis of published materials. First, a set of key words or symbols was identified for each construct. In the research cited, each software quality latent variable was mapped to 4 key words. Keyword frequency was tabulated across a representative sample frame. In the research cited, forty-eight complete papers and articles were tabulated. Categorical statistics and principle component analysis were employed to reduce and prioritize the large base of latent variables. Table 1 summarizes software quality latent variables with their nominal definitions that were identified as most significant in the research cited. These latent variables formed the basis for the survey scales to be presented.

A key finding [ibid] was an affirmation for software of what Garvin [1988] and others have found in the manufacture of hard goods: performance at or above customer expectation levels is the most influential factor for creating an impression of quality. This conformance to stated and unstated expectations was shown [Usrey and Dooley, 1996] to be composed primarily of two elements. Software consumers most frequently expected an abundance of product options or Features. Along with this, a product needed to demonstrate full and Complete development of each aspect of this Feature-rich environment.

The second most critical factor for software product quality was ease of interaction or ease of use [ibid]. Ease of use was also constituted predominantly of two components. The software-to-human interaction aspect is referred to as Simplicity. Interaction between the software and external software and hardware systems is referred to as Communication. Efficiency, the rate of value and waste added per resource consumed, was found to be only a marginal factor in creating a perception of quality. It was however noted to be negatively correlated with both of the ease of use components. In other words, consumers citing Efficiency with a high frequency tended to mention Simplicity and Communication less frequently.

Current utility and future utility are, respectively, the third and fourth most influential factors in creating an impression of software quality [ibid]. Current utility combines the Simplicity construct previously described with predictable software behavior or Stability. Current utility is similar to Garvin's [1988] *perceived quality* and possibly related to first impressions of the product. Future utility combines Flexibility, the marginal cost to extend the product, with a freedom from hardware restrictions classified as Independence.

It has been noted [Usrey and Dooley, 1996] that the task of satisfying customer expectations is far more difficult for the rapidly emerging make-to-stock software market than for make-to-order software which is the subject of the majority of software quality research. Published materials regarding software quality from consumer perspective are primarily in the form of product reviews. The emergence of this market corresponds with the diversification of the software consumer. Humphrey and Sweet [1987], Curtis [1990] and others have suggested Maturity models as a means for characterizing and pursuing quality in the production of software. It seems reasonable that a similar dynamic is at work in the consumption of software. The existence of a continuum of consumer maturity was assumed in both the construction of the cumulative scales and in the development of demographic items.

2.2 INSTRUMENT CONSTRUCTION

Cumulative or Guttman scales offered a promising avenue for operational scale item preparation. Guttman scaling is a "response" method [Torgerson, 1967]. It differs from "judgment" methods in that the systematic variability of subjects to stimuli is ascribed to variability both in the subjects and in the stimuli. In judgment methods, one is primarily concerned with distance and origin of responses. In response methods, we are concerned primarily with the order of responses. From the order, it is possible to postulate and test for a single, underlying continuum called a scale. In Guttman scaling, both survey items (stimulus) and survey subjects (response) are scaled [Tull, 1973]. In most other frequently used techniques, only subjects are scaled.

Preparation of unidimensional scale item requires three steps: discovering scale items, selecting items, and calibrating items [Gorden, 1977]. Four heuristics to discover unidimensional items are sympathetic introspection, participant observation, reading, and interviewing [ibid]. Labaw [1980] warns that introspection and other attempts to understand *why* an attitude exists may be harmful by causing the researcher to produce leading questions. Interviews will be discussed as an aspect of the pilot evaluation and the validation survey. The other heuristics were applied during the preparation for and early phases of this effort.

Hints for selecting items include: cover the full range of attitude responses, avoid asking about past attitudes, work with only one attitude, beware the cognitive-seek the affective and avoid unique experience [Gorden, 1977]. To elicit a full range of attitudes, responses were sought for two software products: one satisfactory and one unsatisfactory.

Calibration of attitude scales helps assure that the full range of attitudes is considered and that many graduations of attitude can be distinguished. Calibration is accomplished on three levels: content, facet, and response. Content calibration is further divided into essence and interaction. Content essence deals with the substance of the attitude and is composed of the criteria of judgment, mutability, and ends-means relationships. Content interaction describes how we relate to objects associated with a given attitude in the form of contact, sanction, and sentiment.

Applying this approach to the Simplicity construct, the positive-essence could be characterized by the statement, “it was simple” and the negative-essence by “it was complex." Facet calibration is a technique to gauge the extent of agreement for the respondent. Five facets of ego-involvement [Gorden, 1977] that have been shown to measure increasing respondent agreement with a candidate attitude are: the societal stereotype, personal beliefs, moral imperative, hypothetical action, and real action (see Table 2).

A societal stereotype deals with the nature or essence of the attitude as viewed by society at large. Personal beliefs are similar, but involve the respondent or an in-group for which they feel representative, like one’s family or department at work. The remaining three facets deal with how the attitude affects interactions. A moral imperative implies how the interactions between the product and an out-group, for example software producers or society at large, should be affected because of the attitude. Hypothetical action is how the individual or in-group would behave or interact with the product because of the attitude. Real action is the respondent reporting actual behavior that has occurred because of the attitude. To illustrate, consider the negative-essence content theme for Simplicity. A potential facet graduation is illustrated in Table 2. Similar facet calibration could be applied to positive-essence and interaction content.

Calibration by response allows for construction of ordinal, interval, or ratio scales within a content-facet pair. Response calibration includes amount of agreement, comparative responses, predictions of action, predictions of feelings, reporting of own actions, threshold of reaction, and strength of response. The “Real Action” facet usually calls for dichotomous responses in reporting of actions. Another popular response calibration is a five point Likert scale ranging from “Strongly Disagree” to “Strongly Agree." Dichotomous response with a “No Response” option was elected for this research effort.

The authors first created a set of scale items including at least one item for each combination of contents and facets. The survey items were then evaluated against objective and subjective criteria: items should avoid complex, confusing, ambiguous or leading words, culturally derived meanings and unclear plurality, minimize questionnaire length [Labaw, 1980]. The set of items selected for the Simplicity construct and their reversal codings are shown in Table 3.

Demographic items were included in the instrument to help determine whether or not a biased sample of software consumers was surveyed. Demographic data can also provide qualitative information regarding instrument validity. Demographic questions included two primary categories of information. Personal characteristics included items such as education, hours of computer use per week, and years of computer use. Software product characteristics include years of experience with the product and computer platform.

Berdie et al. [1986] describe several considerations for survey instructions. With the administration of the survey on the Internet, the right-to-refuse is implicit. Instructions in the software quality instrument address additional considerations as well. Color and bold fonts were used to emphasize key instructions. To increase the response rate within the validation sample, the instructions included appeals to altruism and social utility. Response confidentiality was also assured and respondents were given the option anonymous response. Sharing of research results was also offered to increase response rate. Individuals excluded from the validation sample were identified.

The questionnaire was subjected to a pilot evaluation. The instrument was administered to a select group of five individuals, industry-based experts in the area of software engineering, who were then interviewed. The conditions of the pilot simulated, as nearly as possible, the conditions planned for the actual questionnaire. Interviews focused on possible survey improvements in the areas of survey instructions, questionnaire items and demographic factors that might influence responses. Though the pilot group was too small to provide statistically significant feedback, several valuable qualitative observations resulted from the pilot. After completing the survey, pilot respondents were encouraged to make open-ended statements regarding strengths and weaknesses of the individual items or the survey as a whole. Some survey items were modified to remove jargon. Additional demographic categories were identified. Minor adjustments were also made to the instruction and submission verbiage. The pilot also provided an opportunity to test tabulation, reversal key and analysis routines.

3 INSTRUMENT ADMINISTRATION

Gorden [1977] recommends at least 100 responses for scale construction, so survey economics were in important research consideration. Computer mediated communication is an increasingly important tool to improve the productivity of social research [Lee, 1995]. Some recent examples on-line survey methods are provided by Martin and Manners [1995] and Cockburn and Wilson [1996].

Traditional mailed questionnaires can be solicited or unsolicited. Solicitation and paid responses increase direct costs for each response, but reduce reproduction and transmission costs by improving response rates [Berdie et al., 1986]. Like local call phone surveys, reproduction and transmission costs are relatively insensitive to response volume with Internet-based surveys. These costs are shared by both the researcher and the respondent.

On-line surveys, like mailed questionnaires, enjoy some benefits over other forms of empirical research [Backstrom and Hursh-Cesar, 1981]. There is no bias in the delivery of the instructions. The respondents have as much time as they choose to reflect on their answers. Respondents may also have access to records that would not be available during an interview or other form of empirical research. On-line surveys have additional advantages of cost savings, ease of editing and analysis, faster transmission time, higher response rate, more candid responses, faster responses, and wider coverage [Thach, 1995]. It was noted that problems with response legibility and invalid responses were also avoided. The mechanics of survey return delivery was simplified for the respondent by avoiding the need for return envelopes and postage.

The survey was developed for delivery as a World-Wide Web page using the Internet and the Hyper Text Markup Language (HTML). The survey stimulus was developed using a commercial word-processing package to take advantage of spelling and grammar verification tools. Instructions, scale items and demographic items were all constructed using this technique. The survey files were then saved in rich text format (RTF) and converted to HTML using a public domain software package. The survey (after purification) can be found in Table 10.

The HTML “SELECT” tag was used to interactively record reader responses to scale items. Each response is an enumerated type variable. In the research presented, valid response values were “AGREE," “DISAGREE," and “N/R” for “No Response." Most of the demographic items also used the SELECT tag for responses. “TEXT” type input forms were employed for the remainder of the demographic questions. A “RADIO” type form was used to offer the respondent a copy of the research upon completion. These interaction tags were inserted into the HTML using an ASCII editor. A commercial library of common gateway interface (CGI) utilities was used to process the form upon submission. Hidden input form tags to interact with the CGI process were also entered using an ASCII editor. The survey source code was validated to HTML 2.0 specifications to permit use with a broad array of browsing software. The page was tested with then-current versions of Mosaic, Navigator, Explorer and Lynx browsers.

The software quality survey responses were sought on the World-Wide Web while conforming to Internet etiquette or “netiquette” which discourages unsolicited correspondence [Kroll, 1994; Fraase, 1995]. On-line promotion techniques can be characterized as active or passive [Ellsworth and Ellsworth, 1994].

Passive promotion techniques have a very low risk of offending other on-line users. A user will only encounter passive information if they are engaged in some sort of searching activity themselves. An example of passive promotion is putting the Universal Resource Locator (URL) for the survey to the authors’ “finger” information. Finger is an Internet application designed to give a remote user information about another user. In the UNIX operating system, the syntax for seeing another user's finger information is "finger user@domain" [McGilton and Morgan, 1983]. A UNIX user can append information to the system-supplied finger response by creating an ASCII file ".plan" in their home directory containing the additional information. Table 4 lists additional passive site promotion techniques employed on this project. The mail, news, and world wide web applications involve adding promotional information to the banner and signature footers of electronic messages; a posting was also made to newsgroups.

Active promotions can be troublesome for a user and for their system administrator if the promotions are not conducted in a sensitive manner. Unsolicited promotional e-mail is clearly a violation of netiquette. Some USENET news groups are expressly chartered for announcements, but many groups discourage the unsolicited posting of an advertisement for a product or survey [Canter and Siegel, 1994].

If such an active posting is made, it is considered good "netiquette" to offer readers and respondents compensation. Compensation can take the form of gifts, the chance to enter a contest, or information of interest to the reader. In some cases, the compensation is promoted and the survey is optional if the reader finds value in the compensation and is interested in the survey topic. This approach helps avoid false or nonsense responses aimed at bypassing the survey to obtain the compensation. This was the approach taken in the research presented.

The research project budget didn't include provisions for monetary or tangible remuneration, so WWW-based information servers were developed to attract and benefit survey respondents. For example, *WebPAL*, a list of URLs to facilitate Internet publishing was made available on WWW. The site was promoted using many of the same passive techniques described above, plus announcements in appropriate USENET news groups. Individuals who visited the WebPAL site were asked to complete the survey if they found it to be useful. Table 5 provides a complete list of WWW sites developed for survey promotion and compensation.

4 INSTRUMENT ANALYSIS

The central purpose of the Guttman scaling method is to test unidimensionality, and the primary measure of unidimensionality is the coefficient of reproducibility [Gorden, 1977]. A coefficient of reproducibility of at least 90% is a minimum requirement [Torgerson, 1967; Gorden, 1977].

The coefficient of reproducibility measures the percentage of “error free” responses based on the total responses within a sample. To illustrate the concept of response errors, consider once again the dichotomous Simplicity scale items in Table 2. An individual respondent might agree with the *Societal stereotype*, *Personal beliefs*, and *Moral imperative* statements, but disagree with the *Hypothetical action* statement ”I would recommend against `purchasing the product based on how difficult it is to understand.” Assume that this same respondent agrees with the *Real action* statement; this result is inconsistent with the hypothesis that the five statements form a unidimensional scale and represents a scaling error. Assume that a second individual is unresponsive for the S*ocietal stereotype* statement and disagrees with the other four statements in the scale. There are no scaling errors in the second response set. Taken together, these two response sets would represent one error out of nine total responses or an 8/9 = 89% coefficient of reproducibility. As previously stated, 100 sets of responses is typically sufficient to validate a Guttman scale.

The survey promotion techniques mentioned in the previous section yielded 104 responses for each of the construct scales. This represented about 30% of the actual visits to the web site. The coefficient of reproducibility was calculated for the eight constructs. The values ranged from 88% to 97% with two constructs falling below the 90% threshold (Table 6).

Scales may fail to achieve the 90% threshold for several reasons. Items may not scale because of outliers among the responses, because the hypothesized attitude does not exist, because the items do not reflect the attitude, or because the items are improperly calibrated [Gorden, 1977]. Outlier analysis is used to evaluate the elimination of responses. Techniques to improve item calibration [Torgerson, 1967] include: combine items, eliminate items, combine categories, eliminate categories, and eliminate responses. Factors that influence combining and eliminating items include scale position, item distance, item association, item error and number of items [Gorden, 1977]. Combining and eliminating categories is not applicable in this instance since the survey was designed with dichotomous responses. Taken together, these techniques are sometimes referred to as scale purification [Gorden, 1977].

To illustrate outlier analysis, we turn again to the Simplicity construct. Approximately 2% of the responses displayed patterns with greater than 25% error. These are responses that form saw-toothed or other exaggerated non-scale patterns. Figure 1 shows the relationship between response error and reproducibility for Simplicity. The solid (thin) line depicts responses sorted with percentage of error increasing. This line shows that over 70% of the responses displayed a perfect scale pattern. As responses with pattern errors are included in the calculation, the dashed (thick) line representing scale reproducability descends from 100% to 94%. The two outliers are represented on the extreme right of the “Response Error” line. It is reasonable to assume that other factors such as misinterpretation, inattention, or maliciousness could account for these response patterns.

In empirical modeling with continuous variables, the next step typically taken is the calculation of Pearson’s correlation coefficient. This step is taken to determine if two or more seemingly significant factors are perhaps indicative of a single underlying cause or explanatory variable. For nominal variables, the notion of linear correlation is not meaningful and alternate measures of association are required [Agresti, 1990]. These measures of association should not depend on the order in which categories are listed or upon assumptions about the underlying distributions of the independent and dependent variables [Liebetrau, 1983]. Yules’s Q is a non-parametric measure of monotone association used frequently with nominal variables [Gibbons, 1993]. Q is a useful diagnostic tool for the assessment of dichotomous-item Guttman scales since inconsistent signs indicate possible scale shortcomings [Goodenough, 1944; Torgerson, 1967]. For the sorting and reversal key conventions used in the current study, negative association values are expected.

Remaining with the Simplicity construct as an example, Table 7 shows values for Q among the scale items. These values are calculated after removing the two outliers. The matrix is symmetric, so only those elements below the diagonal are shown. For a set of perfect scale responses, there should be no positive numbers in the association matrix; every "step" along the scale is a step in the same direction. The pattern of positive values in column one suggests further analysis of the first scale item.

One explanation for this association pattern would be that the item represented by column one is an inconsistent indicator of attitudes toward software simplicity. An alternative explanation is that the item, which was intended to convey a societal stereotype, conveys a stronger attitude such as a moral imperative. If we assume that to be the case and resort the scales accordingly, then reproducibility for the Simplicity construct would remain .93 with the association matrix as shown in Table 8. Similar scale purification was performed for each of the constructs. Table 9 summarizes the results. The purified form for all of the survey items is in Table 10.

Demographic data included two categories of information. Information about the individual that was invariant across both software packages was collected and compared, whenever possible, to US population statistics. Information unique to the software package and it's operating environment was captured pairwise (Table 11). Both were collected to determine if any bias was present in the sample that might invalidate the results.

The average age of a survey respondent was 33 compared with the average age of 37 for US citizens. The US average includes all citizens while the survey was biased against those under 18, so a difference of more than four years may in fact be present. However, the survey age data had a standard deviation of 9 years and the US population data appears to be at least as dispersed, so the difference between the survey and the US population data is not statistically sufficient to indicate bias in the software survey.

The percentage of male respondents was 79% for the survey compared with 49% in the US population. This does not necessarily invalidate the survey sample. The difference is likely due to gender bias among Internet users. An article in *Time* [September 4, 1994] suggests that up to 95% of Internet users are male, although more recent data indicates a reversal in that trend. This is a result of societal factors that make males more likely to participate in career activities involving computers, which in turn make them more likely to purchase computer hardware, software and on-line services for home use. This same bias is properly reflected in the software survey data.

Educational achievement data for the survey sample also vary significantly from US population norms. An argument similar to the one used for gender bias is certainly relevant here. Higher educational achievement makes one more likely to interact with computers on the job and at home. Among those who have achieved college degrees, the relative frequencies are similar in both the US population data and the survey sample. This suggests that no significant additional bias exists. The remaining individual demographic statistics do not have readily available reference distributions for comparison.

The pairwise demographic data are also generally supportive of the claim of an unbiased sample. Though the survey protocol prohibits disclosing the names of the software packages cited by the survey respondents, numerical characteristics of the sample are still informative. Out of 104 scale responses, there were 75 distinct software packages cited and only 13 were mentioned more than once. This suggests that respondents came from a diverse consumer population and an unbiased sampling process. Of the thirteen mentioned more than once, 6 were mentioned only favorably, 4 were mentioned only unfavorably, and 3 were cited in both categories. The fact that packages were cited by different respondents in both favorable and unfavorable categories supports the notion that quality is a function of both the product and the consumer.

The ratio of business to personal use software is nearly identical for both favored and disfavored packages. Likewise, the preponderance of make-to-stock software observations is present in both sets of software. Comparison of the platforms used was limited to visual inspection, but suggests no bias that might account for the survey results. Favored software is owned 20% more often than disfavored software. This existence of this difference was not surprising. It seems reasonable to expect that an individual is more likely to buy a favored product than a disfavored one. Only a third of the consumers surveyed received training on their favored software with an 18% increase for training on disfavored software. It is unclear whether this indicates that training is a poor mechanism for increasing software satisfaction, or that shortcomings were recognized in the disfavored package and training was used in an attempt to overcome those shortcomings. Finally, the survey group used the favored software for an average of 2.3 years, 44% longer than the average for disfavored software. One interpretation is that familiarity breeds favor. If so, then continued use of the disfavored software would lead to that package eventually earning favor. A more likely interpretation is that favored software packages earn the loyalty of their users and increased longevity.

5 INSTRUMENT VALIDATION

The scales were intentionally developed applying information for a broad variety of software packages and user demographic characteristics. Further scale validation was desired measuring a more homogeneous group under more controlled conditions. This also presented an opportunity to display a practical application of the scales. The purified instrument was delivered to group of university students enrolled in the course *Design and Analysis of Experiments*. The students used two different software packages to solve a problem. The validation sample was too small to provide statistical validation, but provided important qualitative feedback on the strengths and weaknesses of the scales.

To control the purpose for which the software packages were being used, the students were given a specific task. They were required to confirm the calculation of the effect coefficients of a 2^3 full factorial experiment presented in the course text [Box, Hunter, Hunter; 1978]. The coefficients of the particular experiment were to be calculated using two different statistical analysis software packages. The authors subjectively submit that “Package 1” is a far superior software product for solution of the given problem. These packages would then be compared using the instrument scales.

To control for any previous experience the respondents may have had, a demonstration of the two packages was given. First, coefficients for an example 2^3 experiment were manually derived. Then a computer and projection system were utilized to show the calculations using each of the two candidate software packages. The presentation time for each package was equal and constrained to minimize the opportunity for biased delivery by the presenter [Cannell and Kahn, 1968]. No other instructional activity involved the packages until after the homework assignment was due. Immediately after the homework due date, the scale responses were gathered.

For the Simplicity construct, the reproducibility in the validation group was .97. The errors were uniformly distributed among items and respondents, so none was excised from the sample. The association matrix for the validation group (Table 12) provided an unexpected result: The pattern of monotonicity was a fairly erratic.

During investigation of this result, an interesting response pattern was noted. Sorting the coded responses by the respondent and then by the sum (Table 13) indicated a strong bias within individuals in the survey group. Rather than measuring the difference between software packages, the survey apparently measured some characteristic in the students.

Four respondents found both software packages to be overly complex, while five students found both packages easy to use. The response of one student was ambiguous, with a generally favorable rating for “Package 2," but failing to respond for the competitor on most items. In a follow-up discussion with the student, it was indicated that the both software packages were equally easy to use. Survey time was used in preparation for another class.

One characteristic that might account for these results is the students’ facility with the problem domain of solving for the coefficients in a factorial experimental design. Students that were comfortable with the problem domain might be easily satisfied with either product, while students with a poor grasp of the problem domain are more easily confused by either software package. Another explanation is the differential between the response and the true attitude [Labaw, 1980]. For example, the respondent produces the answer they think the researcher wants to hear or responds maliciously. Respondent cognition is another concern [ibid]. The controlled circumstances of the validation survey may not have allowed the individuals an opportunity to develop the nuances of attitude that were sought to measure.

The other notable inconsistency in the validation survey was identified while analyzing the Communication construct. The reproducability coefficient of 83% in the validation survey was significantly smaller than the 92% originally observed. The association matrix indicated that the majority of the pattern errors occurred on the items associated with action facets. Gorden [1977] suggests that, instead of indicating an indifferent attitude, this pattern can be emblematic of respondents that have not had an opportunity to perform the action. For example, consider the item “I have discouraged others from purchasing the product based on its poor ability to import and export information with other products.” A negative response can denote either dissatisfaction with the Communication aspects of the product or no occasion to discourage another. The students included in the validation survey had only two weeks between introduction of the software packages and the survey, so may well have lacked occasion to perform some of the action facets represented in the survey.

6 CONCLUSIONS

Ordinal scale techniques appear to be well suited to certain aspects of software quality measurement. Several key dimensions of software quality appear to have highly reproducible survey results for the types of software evaluated.

Several of the benefits cited for on-line survey techniques were realized in the current research. The technological experience of many software consumers offers a nearly ideal environment for the application of on-line technologies, especially considering the large sample sizes required for scale construction.

Derivative research efforts could repeat the survey methodology described herein in an attempt to replicate the results. This would provide strong evidence for both the reliability and validity of the current research. The scales could also be applied with a more narrowly focused market group, similar to the validation sample but larger in scale. This would illustrate practical applications for the basic research and development presented here.

7 ACKNOWLEDGMENTS

This work has been partially supported by Protocol Communications, Inc. We wish to thank the many volunteer participants who completed questionnaires during the pilot, survey and validation stages of the research. We also thank editor Andreas Schramm.

BIBLIOGRAPHY

Agresti A., *Categorical Data Analysis*. New York: John Wiley & Sons 1990.

Albrecht J. , "Measuring Application Development Productivity", *Proceedings of IBM Applications Development Symposium*. GUIDE International, October 1979.

Anastasi, A.: *Psychological Testing*, 6th ed, New York, MacMillan, 1986.

Babbie R. , *The Practice of Social Research*, 2nd edition. Belmont: Wadsworth, 1979.

Backstrom H. and Hursh-Cesar G.: *Survey Research*, 2nd edition. Wiley, 1981.

Berdie D. R., Anderson J.F., Niebuhr M.R.: *Questionnaires: Design and Use*, 2nd ed, Scarecrow Press, 1986

Boehm W. , "Verifying and Validating Software Requirements and Design Specifications", *IEEE Software*. New York: IEEE Computer Society Press, January 1984.

Boehm W., Brown J. R., Kaspar H., Liplow M., Macleod G. J., and Merrit M. J. , *Characteristics of Software Quality*. Amsterdam: North-Holland, 1978.

Box, GEP, Hunter, WG, Hunter, JS, *Statistics for Experimenters*, New York, Wiley, 1978.

Bucci, G.: "The Impact of Software Quality," *Objective Software Quality*. Proceedings Second Symposium on Software Quality Techniques and Acquisition Criteria, Florence, Italy May 29-31, 1995, ed. Nesi, P., Berlin: Springer Verlag, 1995

Cannell F. and Kahn R. L.: "Interviewing", *Handbook of Social Psychology*, 3rd Edition (Lindzey, editor). Reading, MA: Addison-Wesley, 1985.

Canter, L.A., Siegel, M.S., *How to Make a Fortune on the Information Superhighway*, HarperCollins, New York, 1994.

Cho C. K. , *Quality Programming: Developing and Testing Software using Statistical Quality Control*. New York: Wiley, 1987.

Cicu, A.: "Process Quality Problems from the Point of View of the User and the Purchaser of Software," *Objective Software Quality*. Proceedings Second Symposium on Software Quality Techniques and Acquisition Criteria, Florence, Italy May 29-31, 1995, ed. Nesi, P., Berlin: Springer Verlag, 1995

Cockburn, C., Wilson, T.D.: "Business Use of the World-Wide Web," *International Journal of Information Management*, Volume 16, Number 2, ed. Hills, P., Exeter, GB: Elsevier, April 1996

Coletta, T.: "The SPICE Project: An ISO Standard for Software Process Assessment, Improvement and Capability Determination," *Objective Software Quality*. Proceedings Second Symposium on Software Quality Techniques and Acquisition Criteria, Florence, Italy May 29-31, 1995, ed. Nesi, P., Berlin: Springer Verlag, 1995

Corbett, M., Porteous, M., Kirakowski, J.: "How to use software usability measurement inventory: the user's view of software quality," *Proceedings of the European Conference of Software Quality*, Proceedings of the European Conference on Software Qualtiy, November 3-6, 1992, Madrid.

Crosby, P.: *Quality is Free: The Art of Making Quality Certain*, New York, McGraw-Hill, 1979.

Curtis B, Sheppard S. B., Milliman P., Borst M. A., and Love T. , "Measuring the Psychological Complexity of Software Maintenance Tasks with the Halstead and McCabe Metrics", *IEEE Transactions on Software Engineering March 1979*. New York: IEEE Computer Society Press, 1979.

Curtis B, "Managing the Real Leverage in Software Productivity and Quality", *American Programmer*, Vol 3, No 7-8, Arlington, MA: Cutter Information Corp, July/August 1990.

Department of Defense , *DOD-STD-2167 Defense System Software Development NAVMAT 09Y*. Washington DC: Department of Defense, 1985.

Dobbins J,H and Buck, R.D.: "The Cost of Software Quality," *Handbook of Software Quality Assurance.* ed. Schulmeyer, G.G. and McManus, J.I., New York: Van Nostrand Reinhold, 1987

Dreger B. , *Function Point Analysis*. Englewood Cliffs: Prentice Hall, 1989.

Ellsworth, J.H., Ellsworth, M.V., *The Internet Business Book*, John Wiley, New York, 1994.

Fagan M. E. , "Design and Code Inspections to Reduce Errors in Program Development," *IBM Systems Journal*, Number 3. July 1979.

Fraase, M., *The Mac Internet Tour Guide*, 2 ed., Ventana Press, Chapel Hill, NC, 1995.

Garvin A. , *Managing Quality*. New York: Free Press, 1988.

Gibbons, J.D.: *Nonparametric Measures of Association*, Newbury Park: Sage, 1993.

Gibbs, W.W., "Software's Chronic Crisis," *Scientific American*. September 1994, p. 86-95.

Goldberg, R. and Lorin, H.: “On the Economics of Information Processing,” *The Economics of Information Processing*, Vol 1, ed. Goldberg, R. and Lorin, H., New York: Wiley, 1982

Goodenough, “¡ Technique for Scale Analysis,” *Educational and Psychological Measurement*, vol 4, Durham, NC: *1944.*

Gorden L. , *Unidimensional Scaling of Social Variables*. New York: Free Press, 1977.

Guaspari, J. , *I Know It When I See It: A Modern Fable About Quality*. New York: American Management Association. 1985.

Guttman L.: "A Structural Theory for Intergroup Beliefs and Action," *American Sociological Review*, Volume 24, Number 3. June 1959.

Hicks, J.O., Jr.: *Management Information Systems: A User Perspective*, St. Paul, MN: West, 1984.

Humphrey S. and Sweet W. L. , *A Method for assessing the Software Engineering Capability of Contractors, Preliminary Version, Pittsburgh: Software Engineering Institute, September 1987*. CME/SEI-87-TR-23, ESD-TR-87-186, ADA187230, September 1987.

ISO 9000-3:1991: “Quality management and quality assurance standards - Part 3,” *Guidelines for the application of ISO 9001 to the development, supply and maintenance of software*, International Organization for Standardization, 1991

ISO 9001:1994 *Quality systems - Model for quality assurance in design, development, production, installation and servicing*, International Organization for Standardization. 1994

ISO 9004-1:1994: “Quality management and quality system elements - Part 1,” *Guidelines for quality improvement*, International Organization for Standardization, 1994

ISO 9004-4:1994: “Quality management and quality system elements - Part 4,” *Guidelines for quality improvement*, International Organization for Standardization, 1994

ISO 9126, International Standard ISO/IEC 9126, *Information Technology--Software Product Evaluation--Quality Characteristics and Guidelines for Their Use*, International Organization for Standardization, International Electrotechnical Committee, 1991.

Johnson, J.R., *The Software Factory: Managing Software Development and Maintenance*, Wellesley, MA: QED Information Sciences, 1989.

Jones T. C. , *Measuring Software*. Cambridge: MA, Software Productivity Research, 1990.

Jones, G.W., *Software Engineering*, New York: Wiley, 1990.

Karlsson, E.A., and Morel, J.M.: "The Impact of Reuse on Software Quality," *Objective Software Quality*. Proceedings Second Symposium on Software Quality Techniques and Acquisition Criteria, Florence, Italy May 29-31, 1995, ed. Nesi, P., Berlin: Springer Verlag, 1995

Kemerer C. F. , "An Empirical Validation of Software Cost Estimation Models," *Communications of the ACM*. New York: Association for Computing Machinery, 1987.

Kroll, E., *The Whole Internet*, 2nd ed., O'Reilly & Associates, Sebastopol, CA, 1994.

Labaw PJ: *Advanced Questionnaire Design*. Cambridge, MA: Abt Books, 1980.

Lee, R.M.: “Information technology for the social scientist: an introduction,” *Information Technology for the Social Scientist*, edited by Lee, R.M., London: University College London, 1995.

Liebetrau, A.M.: *Measures of Association*, Beverly Hills, CA: Sage, 1983.

Long, L.: *Introduction to Computers and Information Processing*, Englewood Cliffs, NJ: Prentice-Hall, 1984.

Martin, J. and Manners, T.: “Computer assisted personal interviewing in survey research,” *Information Technology for the Social Scientist*, ed. Lee, R.M., London: University College London, 1995.

McCall, J.A., Richards, P.K., and Walters, G.F., *Concepts and Definitions of Software Quality,* Springfield, 1977.

McGilton, H, Morgan, R, *Introducting the UNIX system*, McGraw-Hill, New York, 1983.

Meyer, B.: *Object-oriented Software Construction*, New York: Prentice-Hall, 1988.

Misterek D. A., Anderson J., and Dooley K. , "The Strategic Nature of Process Quality," *Proceedings of Decision Science Institute*. Decision Science Institute, 1990.

Mullen, M.: *Object Oriented Program Design*, Reading, MA: Addison-Wesley, 1989

Paulk, M.C. et al*: SEI-93-TR-025 - Key Practices of the Capability Maturity Model*, Version 1.1, Software Engineering Institute, Pittsburgh, PA

Prell M. and Sheng A. P. , "Building Quality and Productivity into a Large Software System," *IEEE Software*. New York: IEEE Computer Society Press, July 1984.

Ramamoorthy V., Prakash A., Tsai W. T., Usuda Y., "Software Engineering: Problems and Perspectives," *IEEE Computer*. New York: IEEE Computer Society Press, October 1984.

Schach, S. R.: *Software Engineering*, Homewood, IL: Irwin, 1990.

Schulmeyer G. , "Software Quality Assurance -- Coming to Terms," *Handbook of Software Quality Assurance*. New York: Van Nostrand Reinhold, 1987.

Scott, WA: “Attitude Measurement,” *Handbook of Social Psychology*, 3rd ed., edited by Lindzey, G., Reading, MA: Addision-Wesley, 1985.

Thach, L.: “Using Electronic Mail to Conduct Survey Research,” *Educational Technology*, Volume 35, Number 2, Englewood Cliffs:NJ, Educational Technology Publications, March-April 1995.

Time Magazine, Time-Warner: New York, September 4, 1994.

Torgerson, WS, *Theory and Methods of Scaling*, New York, Wiley, 1967.

Tull DS: *Survey Research: A decisional approach*. CITY: Intext Educational Publishers, 1973.

Usrey M. W. and Dooley K. J.: "The Dimensions of Software Quality", *Quality Management Journal*, Volume 3, Issue 3, (Easton, editor). Milwaukee, WI: ASQC, July 1996.
Zeithaml A., Parasuraman A., Berry L. L. , *Delivering Quality Service*. New York: Free Press, 1990.

Biographical Sketches

Mike Usrey is the president of Protocol Communications, Inc. He is a doctoral degree candidate in Industrial Engineering at the University of Minnesota. Usrey may be contacted at Protocol Communications, 43 Main Street SE, Suite 132, Riverplace Exposition Hall, Minneapolis MN 55414; 612-623-9900, Fax 612-623-9902, or email musrey@protocom.com.

Kevin Dooley is an associate professor and the director of the Industrial Engineering program at the University of Minnesota. He earned a doctorate degree from the University of Illinois. He may be reached at 125 Mechanical Engineering, 111 Church Street SE, Minneapolis, MN 55455; 612-625-0077, Fax 612-624-1398, or email kdooley@maroon.tc.umn.edu.

Communication	form of inputs and outputs is easy to assimilate into larger system (B)
Completeness	each part full developed (B)
Features	operating characteristics (G)
Flexibility	marginal cost to extend Features (M)
Independence	executable in hardware environment other than current one (B)
Maturity	application of experience to control and improve process (H)
Simplicity	how complicated (M)
Stability	predictability (M)

Adapted from: (B) Boehm et al, 1978; (G) Garvin, 1988, (H) Humphrey and Sweet, 1987, (M) Misterek et al, 1990.

TABLE 1 quality constructs and nominal definitions as identified by Usrey and Dooley, 1996.

FACET	STATEMENT WORDING
Societal stereotype	Most users would find this product too complicated
Personal beliefs	I found this product too complicated
Moral imperative	The product developers were not concerned with making a easy to understand product.
Hypothetical action	I would recommend against purchasing the product based on how difficult it is to understand.
Real action	I have recommended against the product based on how complex it is.

TABLE 2 negative-essence content facets for Simplicity

Reversal Key	Item
1	Most users would find the product easy to understand.
1	I found the product easy to understand.
1	The product developers seemed concerned with ease of understanding.
-1	I would recommend against purchasing the product based on how difficult it is to understand.
1	I would recommend the product based on how easy it is to understand.
1	I have recommended the product based on how easy it is to understand.

TABLE 3 - selected items and reversal codes for Simplicity construct

Application	**Method**
finger	.plan file
mail	.vacation file
mail and news	.sig file
world wide web	meta tags
world wide web	search engines
world wide web	site indices
news	comp.infosystems.www.announce

TABLE 4: Passive web site promotion techniques

Original Content	**URL**	**Active Promotion**
Engineering and Technology Management Degree Programs	http://www.me.umn.edu/0h/home/musrey/ie.html	WWW Virtual Library, sci.op-research
Technology Management Organization Chart	http://www.protocom.com/protomall/Protocom/bizresrc/	alt.business.career-opportunities.executives, alt.office.management, sci.op-research
Web Publishing Assistance Links (WebPAL)	http://www.protocom.com/protomall/Protocom/bizresrc/WebPAL	comp.infosystems.www.authoring.html, comp.infosystems.www.authoring.misc, comp.infosystems.www.authoring.images, alt.html, alt.html.editors.webedit
The Dimensions of Software Quality	http://www.me.umn.edu/0h/home/musrey/research.html	misc.industry.quality, comp.doc.techreport, comp.software-eng
IEOR 5311 site	http://www.me.umn.edu/0h/home/musrey/5311/syllabus.html	University of Minnesota Mechanical Engineering home page, personal home page

TABLE 5: Active web site promotion

SCALE	REPRODUCIBILITY
Communication	0.90
Completeness	0.88
Features	0.97
Flexibility	0.88
Independence	0.93
Simplicity	0.94
Stability	0.92
Maturity	0.91

TABLE 6 - Raw Survey Reproducibility

1.00					
0.11	1.00				
0.09	-0.02	1.00			
0.09	-0.02	0.00	1.00		
-0.03	-0.15	-0.12	-0.13	1.00	
-0.04	-0.15	-0.13	-0.13	-0.01	1.00

TABLE 7: Item Association (Yule's Q) - Simplicity Construct

1.00					
-0.02	1.00				
-0.02	0.00	1.00			
-0.11	-0.09	-0.09	1.00		
-0.15	-0.12	-0.13	-0.03	1.00	
-0.15	-0.13	-0.13	-0.04	-0.01	1.00

Table 8 - Item Association - Simplicity Construct with Item Reordering

SCALE	REPRODUCIBILITY
Communication	0.92
Completeness	0.92
Features	0.98
Flexibility	0.90
Independence	0.97
Simplicity	0.93
Stability	0.97
Maturity	0.91

TABLE 9 - Reproducibility with scale purification

TABLE 10 - Purified Scale Items

SCALE	REV	ITEM
		Communication
1.01	1	It is important that the inputs and the outputs of this product are portable to other software.
1.02	-1	The current market for this product will tolerate a product with inputs and outputs that don't share information with other software.
1.03	-1	I believe it would be difficult to import and export information with this product to integrate with other products.
1.04	1	The product developers seemed concerned with how easily information is imported and exported with this product.
1.06	1	I believe that it is easy to import and export information with this software product.
1.07	1	I would purchase this product based on the ease with which the inputs and outputs can be used with other products.
1.08	-1	I have discouraged others from purchasing the product based on its poor ability to import and export information with other products.
1.09	-1	I have sought outside help or references to make the product integrate information with other products.
1.10	1	I have purchased this product based on the ease with which the inputs and outputs can be used with other products.

Completeness

2.01	1	I thought that all the existing product elements were fully developed.
2.02	-1	The product developers did not seem concerned with fully realizing all existing functions of the product.
2.03	-1	The current market will tolerate a product with existing functions that are less than fully developed.
2.04	1	Most users would feel that all the existing product elements were mature.
2.05	1	I would purchase this product based on the maturity of product functions.
2.06	1	I have purchased this product based on the fully developed product functionality.

Independence

3.01	1	Most users would upgrade their hardware and/or operating system to use this product.
3.02	1	I would recommend this product even if it was necessary to upgrade hardware and/or operating system.
3.03	1	I would purchase this product even if I had to upgrade my hardware and operating systems.
3.04	-1	I would not upgrade my hardware and/or operating system to use this product.
3.05	1	I purchased this product even though I had to upgrade my hardware and operating system.

Features

4.01	-1	Most users would believe that the product's functions were below expectations.
4.02	1	Most users would believe that the product's functions exceeded expectations.
4.03	1	The product's functions exceeded my expectations.
4.04	-1	I would not recommend the product based on the functions failing to meet expectations.
4.05	1	I have recommended the product because the functions exceeded expectations.

Flexibility

5.01	-1	It is not important that the product is easy to extend and adapt.
5.02	1	I believe that the product's outputs would be easy to adapt.
5.03	1	The developers of the product seemed concerned with making the product easy to extend and adapt.
5.04	-1	I would not recommend developing extensions to the product.
5.05	-1	I am considering discontinuing use of the product because of it is so difficult to adapt and extend.
5.06	1	I could develop adaptations to the product.
5.07	-1	I have discontinued using the product because it is so difficult to adapt and extend.
5.08	1	I have recommended developing adaptations to the product.
5.09	1	I have developed adaptations to the product.

Simplicity

6.01 1 I found the product easy to understand.

6.02 1 The product developers seemed concerned with ease of understanding.

6.03 -1 I would recommend against purchasing the product based on how difficult it is to understand.

6.04 1 Most users would find the product easy to understand.

6.05 1 I would recommend the product based on how easy it is to understand.

6.06 1 I have recommended the product based on how easy it is to understand.

Stability

7.01 1 It is important that the product behave predictably.

7.02 1 I believe that the product behaved in a predictable fashion.

7.03 -1 The product developers seemed unconcerned with making the product behave predictably.

7.04 1 Most users would believe that the product behaved in a predictable fashion.

7.05 -1 It is a problem that the product behaves unpredictably.

7.06 1 I have recommended this product based on its predictable behavior.

Maturity

8.01	1	I considered other competing products.
8.02	1	I sought reviews of the product and competitors.
8.03	1	I established selection criteria for this purchase.
8.04	1	I purchased this product on impulse.
8.05	-1	I have not reviewed my product selection criteria based on experience from this purchase.

Favored Software Package	Disfavored Software Package	
72%	74%	Business Use
98%	94%	Make to Stock
72%	60%	Own Software
34%	40%	Formally Trained
2.3	1.6	Package Experience (Years)
FIGURE 2	FIGURE 3	Platform Distribution

TABLE 11: Pairwise software demographic data

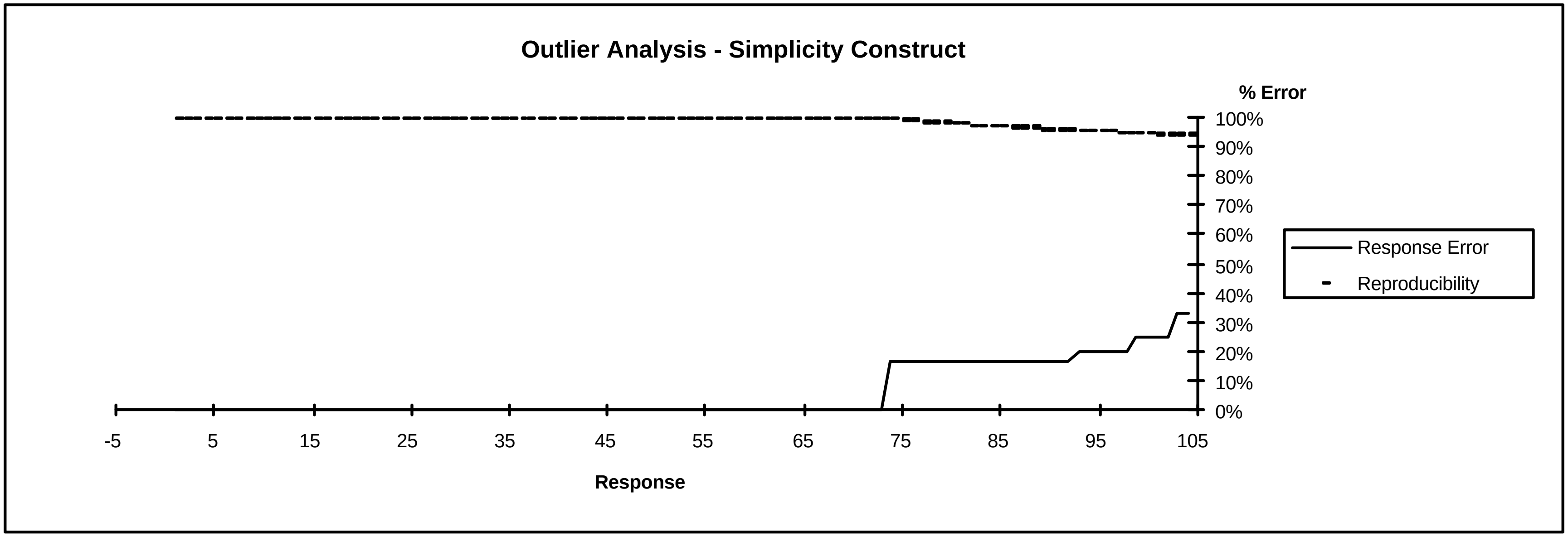
1.00					
0.04	1.00				
-0.05	-0.09	1.00			
0.11	0.07	0.16	1.00		
0.22	0.18	0.27	0.12	1.00	
0.00	-0.04	0.05	-0.11	-0.22	1.00

Table 12 - Item Association - Simplicity Construct for Validation Sample

Respondent	1		2		3		4		5		6		7		8		9		10	
Package	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
Item 1	-1	-1	-1	-1	-1	-1	-1	-1	1	1	1	1	1	1	1	1	1	1	1	1
Item 2	-1	-1	-1	-1	-1	-1	-1	-1		1		1	1	1	1	1	1	1	1	1
Item 3	-1	-1	-1	-1	-1	-1	***1***	-1		1	1	1	1	1	1	1	1	1	1	1
Item 4	-1	-1	-1	-1	-1	-1	-1	-1		1	1	1	1	1	1	1	1	1	1	1
Item 5	-1	-1	-1	-1	-1	***1***	-1	***1***		1	1	1	1	1	1	1	1	1	1	1
Item 6	-1	-1	-1	-1	-1	-1	-1	-1		1	1	1	1	1	1	1	1	1	1	1

Table 13 - Simplicity Construct - Validation Survey Coded Responses

(Underlines indicate response pattern errors)



Outlier Analysis - Simplicity Construct
% Error
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Response Error
Reproducibility
-5
5
15
25
35
45
55
65
75
85
95
105
Response

Figure 1