Quality Engineering 3(1), 27–40 (1990–91)

A METHOD FOR ASSESSING SOFTWARE MEASUREMENT TECHNOLOGY

MICHAEL K. DASKALANTONAKIS
Motorola, Inc.
3701 Algonquin Road, Suite 601
Rolling Meadows, Illinois 60008

ROBERT H. YACOBELLIS
Motorola, Inc.
3701 Algonquin Road, Suite 601
Rolling Meadows, Illinois 60008

VICTOR R. BASILI
Department of Computer Science
University of Maryland
College Park, Maryland

Key Words

Software measurement technology assessment; Fundamental beliefs; Measurement technology maturity themes; Assessment questions; Data analysis and feedback.

Purpose

The purpose of this article is to describe a method for assessing an organization's software measurement technology, as well as the ideas behind this assessment in a way that is consistent with the SEI software process assessment methodology.* Therefore this article:

*The Software Engineering Institute (SEI) is a DoD-sponsored organization affiliated with Carnegie-Mellon University. Its charter is to raise the overall software capability of U.S. industry, particularly for mission-critical software. The SEI has developed a method of assessing the software engineering capability of software development organizations. SEI defined five levels of software development process maturity (1). The highest level (level 5) corresponds to a very capable software development organization. An organization whose capability is assessed, is found to be at one of the five maturity levels at the time of the assessment. SEI has defined a questionnaire (i.e., a set of questions) that is completed by the assessed organization. The answers to the set of questions are used to assign a specific maturity level to that organization.

1. Introduces a re-engineered SEI process assessment methodology which creates a different framework for building a measurement assessment methodology. This framework differs from that of SEI because it focuses on *measurement technology* assessment (instead of development *process maturity* assessment). It is also more concrete and measurable because it tracks the evolution of measurement themes for each measurement technology level.
2. Defines a set of themes associated with the levels of software measurement technology maturity to identify priorities for the assessment of software development organizations, consistent with the SEI process assessment.
3. Provides a method for assessing the current measurement technology level of an organization, consistent with the SEI process assessment with regard to themes and questions. By using this method the organization identifies what is required to improve its measurement technology level.
4. Provides a small subset of the questionnaire used in Motorola to demonstrate the traceability of questions back to themes, the definition of the measurement technology maturity levels, and the consistency with the SEI process assessment.
5. Describes the results of applying the Measurement Technology Assessment at Motorola software development groups.

The following section presents the assumptions on which the "Measurement Technology Assessment" is based. The next two sections define the five levels of software measurement technology maturity and provide sample questions included in the assessment. The next section defines the way that the data are used to assess the measurement technology maturity level. The last two sections provide qualitative feedback from our experiences in using the approach at Motorola and the conclusion of this report.

Assumptions

There is a set of assumptions for defining the levels of software measurement technology maturity and the assessment questions (a second set of assumptions is used for classifying the questions of the assessment by maturity level, and it is described in a subsequent section). The first set of assumptions consists of the fundamental assumptions/beliefs that determine the focus of the Measurement Technology Assessment. The assessment is based on the following beliefs:

1. A well-defined, quality-focused, software development *process* will very likely result in a quality software project and product. The introduction of mechanisms for defect prevention and early detection (e.g., process definition, software reviews, testing, and quality assessment) are significant factors contributing to product and process control. Therefore, the following theme is important:

Theme 1: Formalization of the development process (i.e., existence and use of a documented and approved software development process)

2. *Measurement* is facilitated by, and facilitates, a well-defined, software development process and product. It requires a well-defined process itself which is integrated into the development process. This measurement process includes automation of the data collection; evaluation and feedback of deficiencies; and the improvement of an organization's projects, products, and processes. Tools such as configuration management, as well as problem tracking and analysis, permits emphasis on key measures for tracking the process. The use of a database for tracking project, product, and process data is critical for formally capturing and analyzing knowledge that will be used to characterize, evaluate, and improve the software development process and product. Therefore, the following themes are important:

   Theme 2: Formalization of the measurement process (i.e., existence and use of a documented and approved measurement process)
   Theme 3: Scope of measurement within the organization
   Theme 4: Implementation support for formally capturing and analyzing knowledge

3. There is an *evolutionary pattern* (hierarchy) that measurement follows. We start with project, then product, and finally process measurement. Project data (e.g., project cost, schedule, etc.) have always been tracked (at least implicitly). Organizations start tracking product data in order to quantify product attributes and quality levels. When an organization realizes the impact of a formalized process on the product, it starts tracking also process data and their relationship to product data. Therefore measurement results in accumulation of project-specific data, product data associated with classes of products and process data. These data are used to improve management control of software projects. Therefore, the following themes are important:

   Theme 5: Measurement evolution within the organization
   Theme 6: Measurement support for management control of software projects

4. Project, process, and product *improvement* is achieved by using collected data as information that identify problem areas, and implementing mechanisms for problem prevention based upon informed analysis of the product and process. Improvement can be short-term, based on current-project feedback, or long-term, based on Corporate "memory" of multiple projects and the factors which lead them to succeed or fail. Therefore, the following themes are important:

   Theme 7: Project improvement using measurement technology
   Theme 8: Product improvement using measurement technology
   Theme 9: Process improvement using measurement technology
   Theme 10: Predictability of project, product, and process characteristics

Maturity Levels of Measurement Technology

The measurement of the software development process and product is not a one-dimensional issue (e.g., only scope of measurement). It is multidimensional. A number of aspects influence the software measurement technology maturity. Such aspects include formalization of the measurement process, measurement support for management control, predictability of characteristics, etc. In order to assess and improve the maturity of an organization's measurement technology, we need to specify any important influencing aspects, understand their relationship to the measurement technology, and attempt to control them so that we can impact the measurement maturity level of an organization. Measurement technology improvement cannot be done independently of these aspects, and it is tied to improvements in the software development process.

This provides the motivation for creating the Measurement Technology Assessment and the themes used in this assessment. The ten general themes (principles) listed in the previous section, were derived directly from the assumptions and are used to characterize and evaluate the level of software measurement technology maturity that an organization has reached.

The assumptions and beliefs presented here were stated and the associated themes were derived by

a. Evaluating the SEI process maturity level questions and associated SEI material characterizing the levels, as well as abstracting important concepts hidden in these questions.
b. Stating and refining fundamental assumptions/beliefs and deriving the corresponding themes hidden in these beliefs.
c. Adapting existing themes and developing new ones that cover measurement technology areas (as well as the software development process itself) that required additional coverage (i.e., where Table 1 was inadequate).
d. Generating level definitions for each of the themes, concentrating on consistency among themes in the same level, and completeness of themes.
e. Evaluating how well the themes covered the concept of software measurement technology in the context of the software development process, and appropriately updating the theme table.
f. Generating questions consistent with themes and levels, ensuring traceability of questions to the corresponding themes and levels, and contributing to the repeatability of the question definition and selection process.

As a result, although the themes focus on the measurement of the software development process and product, they extend to the process of software development itself, based on the belief that these two topics are very closely interrelated. For each theme, five evolutionary stages were defined that a software development organization may follow in order to reach the highest level of maturity for that particular theme. These five evolutionary stages correspond to the five levels

of software process maturity as defined by SEI. Using the stages of the themes, it was easier to assess the measurement technology status of a Motorola software development group and track its improvement over time. This resulted in the definition of five levels of measurement technology maturity.

Level i of software measurement technology corresponds to the *i*-th stage of the ten themes that are used to characterize and evaluate the measurement technology maturity of a software development organization. Table 1 precisely defines the maturity levels with regard to these ten themes. The goals of the assessed software development organization should be focused on reaching level five of measurement technology maturity, and they depend on the current maturity level. These goals should not be limited to just reaching the next higher level of measurement technology. A general description of each level (based on the table) follows:

1. Initial: There is no formalization of the software development and measurement process and little or no process definition and measurement is conducted. Measurement is not used to support project management and there is no use of data for problem prevention. Plans and schedules are unstable and commitments are difficult to meet. The process is different for similar projects tending to result in an unpredictable and poorly controlled process (i.e., similar projects tend to have different resource, schedule, and defect profiles). Process and product improvements are impossible to plan and commit to. There is no statistical process control, and no senior management involvement or understanding of the extent of the problem. There are little or no project data and no database is used for storing project data.

2. Repeatable: Process definition and measurement are done at least at the project level and formal procedures are established. Management has basic control of commitments and there is an independent quality assurance process. Organizations display similar results for similar classes of projects. Previously mastered tasks are repeatable. However, the techniques, methods, and processes are not defined to the level that they can be monitored or evaluated and are therefore hard to improve. There is focus on some processes, but the successful use of various processes depends upon the experience of seasoned professionals. There is a disciplined approach to project and configuration management. A database exists for most of the projects for tracking project data.

3. Defined: Product measurement is performed and managed using documented standards. For example, there is a definition of measurement goals, analysis of goals into questions for assessing the progress toward the goals, and definition of metrics for helping the question-driven progress assessment [Goal/Question/Metric paradigm (2)]. These definitions of goals, questions, metrics are packaged together with implementation aids and training materials, and are used within the software development organization. However, the focus is on product rather than process measurement and management. Although there may be a qualitative foundation for applying technology, there are not many quantitative meas-

Table 1. Themes and Levels of Software Measurement Technology Maturity

THEME	LEVEL 1 INITIAL	LEVEL 2 REPEATABLE	LEVEL 3 DEFINED	LEVEL 4 MANAGED	LEVEL 5 OPTIMIZED
#1 Formalization of the development process	Process unpredictable Project depends on experienced and seasoned professionals No/poor process focus	Repeat previously mastered tasks Process depends on experienced people	Process characterized and reasonably understood	Process measured and controlled	Optimized process Focus on process improvement Reward process improvements
#2 Formalization of the measurement process	Little or no formalization	Formal procedures established Standards exist	Documented standards Standards applied	Improvement mechanisms in place Internal standards applied	Organization has learned and improved
#3 Scope of measurement	Done occasionally on projects with experienced people, or not at all	Done on projects with experienced people Project estimation mechanisms exist Project focus	Goal/Question/Metric package development and some use Data collection and recording Existence of specific automated tools Product focus	Metric packages being applied and managed Problem cause analysis Existence of integrated automated tools Process focus	Have learned and adapted metric packages Problem prevention Process optimization
#4 Implementation support	No data or database	Per project database	Product database Standardized database across projects	Process database Common corporate database and process information	Knowledge base Improvement and learning data base
#5 Measurement evolution	Little or no measurement conducted	Project measurement and management	Product measurement and management	Process measurement and management	Continuous feedback and improvement

#6 Measurement support for management control	Management not supported by measurement	Some support by measurement Basic control of commitments	Product measurement and control	Management process measured and controlled Can define quantititive management goals	Can define technology values and needs Can check achievement of goals
#7 Project improvement	No statistical process control No senior management involvement and understanding	Disciplined project and configuration management Risk management Subcontractors evaluated	Dedicated process resources Process data not retained nor analyzed properly Subcontractors controlled	Need discipline to track and eliminate problems Improving technology	Quantitative project feedback Effective reuse
#8 Product improvement	Poor configuration management and quality assurance	Effective (independent) quality assurance Reviews conducted Periodic customer interface for feedback	Qualitative foundation for applying technology Not many quantitative measures of problem causes	Have some mechanisms for determining problem causes	Can analyze problem causes and prevent them
#9 Process improvement	Hard to plan and commit No focus on improvement	Hard to improve the measurement process No foundation for improvement	Quantitative foundation for improvement Training required	Methods and tools tailored to needs Quality improvement Training assessed and improved	Significant productivity improvement
#10 Predictability	Unpredictable results for similar work Unstable plans and schedules	Predictable results for similar work Tools used for project planning	Able to project and track product quality parameters	Process quality and productivity projection and tracking	High predictability due to process formalization and optimization Can analyze problem causes and prevent them

ures for process evaluation and assessment. It is difficult to project and track productivity and quality parameters. The process is characterized and reasonably understood, so that a foundation is laid for quality improvement, but process improvement is difficult because process data are not retained or analyzed properly. There is a product database, which is standardized across projects.

4. Managed: Process and metric packages are being extensively applied and managed. The management process is measured and controlled possibly through the application of internal standards. There are some mechanisms (although not complete) for determining problem causes. Quality parameters can be projected, tracked, and controlled. The process is measured and controlled and there is evidence of quality improvement. The technology is improving but there is still a need for more quantitative feedback on problem prevention. A process database exists, which may have the form of a common Corporate software development process database.

5. Optimized: The organization has learned from the use of the process and metric packages and adapted these packages, in order to optimize the defined process and use better measures. The major organization focus is on improving the software development process based on data from the historical database. Extensive analyses of defect and cost data are conducted. Major defect and cost overrun causes are identified and process improvements are suggested for avoiding these causes. A mechanism exists for initiating defect and failure prevention actions. There is quantifiable process feedback for improvement on the current and future projects. The needs for software technology and the value of existing technology are well-understood. The process can be optimized due to quantitative process feedback and there is significant productivity improvement. The organization as a whole has reached the level that the Productivity and Quality Improvement Paradigm (3) is fully implemented as part of the overall software development process.

Sample Assessment Questions

This section lists sample questions used in the Measurement Technology Assessment. The level number of a question is used for the interpretation of the answers. The questions marked with '#' are taken from the SEI assessment. The questions are formalized in such a way that a "yes" or "no" answer is required. If a question is not applicable, the answer is "no" by default.

An additional set of assumptions was made in order to classify the questions by maturity level. In general, the following guidelines were used:

Questions related to establishing a basic mechanism or process are level 2.
Questions related to data collection and recording, or to the existence of specific automated tools are level 3.

Questions related to data analysis for determining problem causes or to the existence of an integrated set of automated tools are level 4.
Questions related to problem prevention and process optimization are level 5.

At the end of each question a set of numbers is given (within parentheses). These numbers correspond to the themes that justify the level assignment. For example, after the first question in the list, the number 8 appears in parentheses. This means that this question is assigned level 2 because it is related to the second stage of theme 8. Although there has been an effort to ensure consistency, it should be clear that the assignment of themes to questions is not unique. Different people, with different views, may end up with a slightly different assignment of themes. The sample questions listed by maturity level follow:

Level 2: Repeatable

Does the Software Quality Assurance function have a management reporting channel separate from the software development project management? (8)
Is your organization able to repeat previously mastered tasks? (1)
Does a mechanism exist for recording the application domain of each software development project? (10, 5, 7)
Has a managed and controlled project database been established for project metrics data? (5)

Level 3: Defined

Does the software organization use a standardized and documented software development process on each project? (2, 8, 5)
Does the organization have dedicated process resources? (7)
Are defect data found in reviews classified by severity? (1, 9, 5, 6)
Are defect data found in reviews classified by cause? (1, 9, 5, 6)

Level 4: Managed

Is the quality of the work performed tracked against quantitative quality goals? (10)
Is a procedure followed for collecting uniform process data across projects? (5, 1)
Are defects found during testing projected and compared to actuals? (10)
Is there a managed and controlled process database for process data across projects? (4, 5, 6, 1)

Level 5: Optimized

Is a mechanism used to identify and apply lessons learned from each software project for new projects? (5, 2, 8, 10, 6)

Does software quality assurance utilize process data to guide further sampling of the activities and work performed? (10, 1, 5, 2)
Are the causes of defects found in reviews analyzed to determine the process changes required to prevent them? (8, 2, 1, 5)
Is a mechanism used for initiating defect prevention actions? (8, 10, 1, 5, 2)

Data Analysis and Feedback

This section describes how the theme table can be used for providing feedback to the assessed software development organization, based on the answers to the questions of the assessment. The first step in the analysis of the Measurement Technology Assessment data is the identification of the organization's current measurement maturity level. The same maturity level determination algorithm as the one used by SEI was chosen for consistency reasons. However, the importance of determining the number indicating the maturity level itself during the data analysis is minimized (as it is also explained in the next section), in order to emphasize the themes that need improvement.

An assessed organization is at level 2 if at least 80% of all level 2 questions have a "Y" answer. Otherwise, the organization is at level 1.

If the organization is at level 2, whether or not it is at level 3 can then be considered. The same process as the one used for determining whether the organization is at level 2 is used for level 3, etc.

Note that a software development organization may be at a different level of measurement technology maturity than the level of software process maturity determined by the SEI assessment. The reason is that the questions used in the two assessments are not the same. However, there has been an attempt to be consistent with the SEI assessment in the way that the question levels were assigned.

Once the maturity level is determined, improvement themes for the assessed software development organization are identified, based on the assessment questions that had a "N" answer. The themes assist the process of identifying action items. The list of action items should emphasize aspects that are important for reaching the next measurement technology maturity level, but always in the context of eventually reaching level 5. For example, suppose that the assessed organization is found to be at level 2. Suppose also that based on the "N" answers for the questions at level 3, it is determined that one of the themes that needs improvement is "product improvement" (theme 8). The theme table assists us in determining that: qualitative foundation for applying technology and quantitative measures of problem causes, should be the initial focus of the action items, because these items were found important at level 3 of theme 8 in the theme table. However, mechanisms for determining problem causes (level 4 of theme 8), and mechanisms for problem cause analysis and prevention (level 5 of theme 8) should provide additional future direction and the context for the organization's

improvement actions with regards to the "product improvement" theme (i.e., theme 8). The questions associated with these theme stages provide more concrete action items for the organization. As the assessed organization starts defining and deploying additional quantitative measures of problem causes, it also starts defining the mechanisms determining problem causes and initiating preventive actions. This provides an accelerated process improvement focus.

On the other hand, examining other themes, such as "measurement evolution," "formalization of the development process," and "predictability," can give a more balanced view of the kinds of changes the organization needs to make. For example, with regard to "measurement evolution" and "formalization of the development process," it is clear that the process needs to be controlled and process measurement should be established in order to build mechanisms for determining problem causes. It is also clear from the "predictability" theme that process quality and productivity need to be projected and tracked to better understand problem causes. These provide us a more complete picture of the related activities that need to be initiated by the organization to achieve higher levels for the theme "product improvement."

As the Measurement Technology Assessment matures, and its use is being extended within additional groups in Motorola, a more detailed and focused analysis of the data will be conducted. It will be interesting to assess what the organization level is for each particular theme. In this way, additional guidelines for identifying a subset of themes that will have higher priority for improvement will be defined. This means that although it may be determined that an organization is at level 2 with regard to the theme "product improvement," at the same time it could be at level 5 with regards to "measurement support for management control." In a sense, the themes provide an intermediate abstraction between the maturity levels, and the questions that correspond to each maturity level. This continues to be the topic of ongoing research within Motorola.

Experiences from Using the Assessment

Currently, Measurement Technology Assessments have been conducted for three Motorola software development groups. The data were collected through informal interviews with the software managers of the assessed groups. The managers had a good understanding of the software development group organizational structure, and were able to provide answers to the questions of the Measurement Technology Assessment.

The assessment has also been used as an exercise of a workshop on "software metrics" taught internally in Motorola. Although the workshop participants did not always belong to the same software development group, they were able to do a self-assessment for their own groups.

Several benefits have been recognized from conducting the Measurement Technology Assessment within Motorola. The data analysis resulted in identification of problem areas within the assessed organization. Assessment reports have been sent to the managers of the assessed groups indicating these opportunities for improvement. The managers have used these reports to determine what their actions should be. There is an increased interest in effectively using software measurement technology within their organizations. Software measurement functions are created and being staffed to address these needs.

There are several "lessons learned" from conducting the Measurement Technology Assessment within Motorola that are worth mentioning:

1. The process of conducting the assessment should be formalized and a set of steps should be followed in order to ensure the success of the assessment. The interviews for obtaining the answers to the assessment questions should be well-prepared in advance of the meeting itself. The preparation includes the explanation of the assessment's terminology to the software managers, so that it is clear to them what the assessment questions ask them. Terms such as "dedicated process resources," "managed and controlled database," etc. require additional clarification. Conducting the assessment through personal interviews (as opposed to conducting it long distance) is critical to success. The creation of a "glossary of terms" helps resolve any ambiguities. The analysis of the assessment data and the feedback to the group (in a timely manner) assure that the group benefits from this process.
2. The assessment is best conducted when it is focused on a group of similar "software development projects," as opposed to having to assess a group of projects that have different software development and measurement organizational structures. The assessment results can then be abstracted to identify problem areas of the organization as a whole.
3. The focus of the assessment should not just be on determining the number indicating the measurement technology maturity level. Rather it should be on the identification of themes that need improvement. This helps provide an improvement focus for the assessment, instead of viewing the assessments as an audit mechanism. There are cases that the assessed group can be at different maturity level with regards to different themes. The example included in the preceding section explains how this information can be used to identify problem areas and action items. The determination of the maturity level should not be based solely on the quantitative approach used above. A more qualitative focus (based on the themes) is needed.
4. Allowing quantification of the answers given to the assessment questions (as opposed to just allowing yes/no answers) will more likely provide the opportunity of a more detailed assessment. This requires a modification of the current Measurement Technology Assessment. A table similar to Table 1 can be created *for each question of the assessment* to indicate the appropriate

answer within the answer range. This technique has already been used (successfully) within Motorola for other assessments.

These "lessons learned" are currently being considered for updating the Measurement Technology Assessment and making it even more beneficial to the assessed organizations.

Conclusion

In an attempt to define the maturity levels of measurement technology, a table of important themes was constructed which characterizes the measurement maturity level of an organization. During this process, additional themes were identified (and added to the theme table) which are also important for determining the maturity level. These themes were identified based on our fundamental assumptions for defining the measurement technology maturity levels and determining when an assessed software development organization has reached a specific maturity level. The table served as the basis for determining the questions that are appropriate for the Measurement Technology Assessment. The analysis of the data from this assessment results in the definition of a more precise overall focus and a set of action items for an organization to improve its maturity level.

Measurement Technology Assessment has been used to assess the maturity level of a small number of Motorola software development groups with several beneficial results. The "lessons learned" from conducting these assessments indicate that the use of themes for defining the maturity levels has been successful. The use of the themes for defining the measurement technology maturity level provides the basis for understanding the maturity levels, the traceability of questions to levels, and the ability to focus on improvement areas.

The initial results of the analyses conducted in Motorola indicate that the identification and placement of theme stages at the five maturity levels was right for the purpose of the assessment. More precisely, if an organization is assessed as being at level i, the preliminary data indicate that the distance of the organization from maturity level $i+1$ is smaller than the distance from level $i+2$, etc. This means that the themes and their mapping to maturity levels are representative of the stages that an organization should reach. It also indicates that the maturity levels have been defined (through the themes) to be consistent with the degree of difficulty for reaching each evolutionary maturity stage of a software development organization.

This "theme approach" has been expanded to other Motorola assessments (e.g., an assessment of conformance to Motorola's Quality Policy for Software Development) with great success.

The assessment results have been very useful in identifying themes that need improvement throughout the organization, and providing a plan of action for improvement. The use of the assessment has provided goals and motivation for

the software developers for improving their organization's software development and measurement process. This should result in increased productivity within the assessed software development organizations.

References

1. W. S. Humphrey, D. H. Kitson, and T. C. Kasse, The State of Software Engineering Practice: A Preliminary Report. CMU/SEI-89-TR-1, February 1989.
2. V. R. Basili and D. M. Weiss, A Methodology for Collecting Valid Software Engineering Data. *IEEE Transact. Software Eng. SE-10*1(6), 728-738 (1984).
3. V. R. Basili, Quantitative Evaluation of Software Methodology. In *Proceedings of the First Pan-Pacific Computer Conference, Melbourne, Australia, September 1985;* also available as Technical Report TR-1519, Department of Computer Science, University of Maryland, College Park, MD, July 1985.

About the Authors: The authors have participated in the Motorola Metrics Working Group which has the responsibility of providing leadership and internal consulting to Motorola groups in the area of software metrics. The work presented in this paper has been done as part of the activities of this group. Motorola product groups define their software development process and use software metrics to track progress toward their goal of six sigma quality for software. In 1988, Motorola received the first annual Malcolm Baldrige National Quality Award for its successful efforts to achieve high product quality and total customer satisfaction.