Quality Report SQAS95-001

Planning for a Software Process Assessment

May 1995

Software Quality Assurance Subcommittee
of the
Nuclear Weapons Complex Quality Managers

United States Department of Energy
Albuquerque Operations Office

Abstract

This report presents a guide for how to plan for a software process assessment and how to use the assessment results to guide process improvement. It defines the steps associated with conducting a successful assessment and identifies the framework needed to establish a successful process improvement program. The assessment methodology is based upon the Software Engineering Institute's (SEI) software process assessment.

This page is left intentionally blank.

Planning For A Software Process Assessment

Executive Summary

Objectives, Goals and Benefits

Appendices

Assessment Alternatives

Additional Information

Planning For A

Software Process Assessment

Executive Summary

Software process improvement starts with a need by individuals or organizations to improve their software processes. But, just identifying the need is not enough. A method to obtain the desired process improvement must be found. The intent of this paper is to introduce one method for identification of process improvement opportunities. The goal is to provide a method to measure the maturity of software processes and to help evolve them to a desirable status that produces software within the constraints of costs, schedule, and quality.

The assessment methodology introduced is based upon the Software Engineering Institute's (SEI) software process assessment. It is a well documented industry verified and accepted measure of an organization's software capability. It is based upon the SEI Capability Maturity Model (CMM) which can be used for both software process assessment and improvement. Using the CMM as a model, the SEI assessment process provides an objective method to measure an organization's software capability.

This planning document has two main objectives:

1. To help define the steps needed to conduct a successful software process assessment;
2. To identify the framework needed to establish a successful process improvement program.

It achieves those objectives by introducing two software assessment methods based upon the SEI model:

- A self–assessment conducted by the software organizations or software development teams.
- A formal assessment conducted by experienced SEI assessors such as a licensed SEI vendor.

Benefits obtained from applying a software process assessment are addressed, including mapping of the Key Process Areas against representative Department of Energy and international quality requirement documents.

This page is left intentionally blank.

1. Introduction

Software process improvement starts with a need by individuals or organizations to improve their software process. Some reasons as to why software process improvement is important are:

- Software is critical to the future of the Department of Energy and the Nuclear Weapons Complex as a whole.
- Our products are becoming increasingly more software intensive and, at times, the product is the software itself.
- Our customers are demanding software that is correct, on time and within budget. They expect a mature, stable, predictable, managed software development process.
- Management is asking *Is the software correct? Will it be done on time? Is it within budget? How good is our software process? Could the software be done better and faster?*

But, just identifying a need is not enough. A method to obtain the desired process improvement must be found. The intent of this paper is to introduce one method to do that. It helps an organization get started by providing guidance and help. Organizations may choose to implement all or part of what is presented here. The paper's goal is to provide an organization with one method to understand their software process and help evolve their process to one that produces software within the constraints of costs, schedule and quality.

This page is left intentionally blank.

2. Objectives and Goals

The intent of this paper is to provide and promote a standard and structured approach to establishing software process improvement. It is based upon the accepted method of improving any process:

1. Know your current process and problems.
2. Envision and define the target process you would like to have.
3. Plan and implement changes from the current to the envisioned.
4. Repeat steps 1-3.

Additionally, the improvement process works better if the target process can be measured against an accepted standard. This paper addresses all those issues but its main emphasis is on defining a software process assessment method where by an organization can understand their current process and problems.

The software process assessment introduced here is based upon the Software Engineering Institute's (SEI) software process assessment. It is a well documented industry verified and accepted measure of an organization's software capability. It is based upon the SEI Capability Maturity Model (CMM) which can be used for both software process improvement and software process assessments.

A process is a set of actions, tasks and procedures that when performed or executed, obtain a specific goal or objective. More specifically, a software process is a process whose goal is to manage the development/maintenance of software. A software process assessment is an appraisal or review of that software process.

Using the CMM as a model, the SEI assessment process provides an objective method to measure an organization's software capability. The assessment process establishes a baseline of software capability from which to measure future improvement. In the words of Watts Humphrey, the architect of the Capability Maturity Model, *"If you don't know where you are, a map won't help."*

It also helps:

- Identify software productivity and quality goals for each organization for "*If you don't where you are going, any road will do."*
- Establish the beginning of a customized road map to software process improvement to achieve those goals.

This paper has two main goals:

1. Help define the steps needed to conduct a successful software process assessment.
2. Identify the framework needed to establish a successful software process improvement program.

It achieves those goals by introducing two software process assessment methods based upon the SEI model. The outcome of the assessments is the characterization of the current software process and a set of improvement recommendations. The objectives of both assessment methods are the same, but the formality and the details vary.

This page is left intentionally blank.

3. Benefits

There are numerous benefits to be obtained from the application of software process assessments. The focus of an assessment is on where you are as an organization in software development or maintenance capability, but the overall benefit is that by identifying problem areas, an assessment provides a method to achieve improvement. Then you can chart a course to get to where you want to be. This section outlines some of the more tangible, measurable benefits.

The underlying Software Engineering Institute's (SEI's) assertion is that improving individual software process areas improves the level of the organization's capability. The assessment of the current software process areas against the SEI's Capability Maturity Model (CMM) determines where changes can provide the most benefit. Metrics must be developed for each of the benefits outlined in this section. Ideally, a baseline measurement should be established prior to application of the assessment and prior to implementation of action plans.

3.1. Cost Benefits

Software companies with successful improvement programs have reported that their return on investment (ROI) is typically 6:1. The primary cost benefits might include:

- Lowered cost of software maintenance.
- Improved cost of software quality (fewer defects, less rework).
- Lowered cost of ownership for equipment with embedded software.

The cost benefits will not be achieved until action plans resulting from the software assessment process are implemented. The areas must be carefully chosen to produce the cost benefits.

An additional cost benefit is the cost–avoidance of re–inventing/developing these assessment procedures. That is, by employing these assessment procedures, organizations desiring to discover how their processes stack up as measured on a national scale will be leveraging the development efforts of the Software Engineering Institute, and the implementation results reported in numerous journals.

3.2. Motivational/Staff Benefits

In general, motivation benefits answer the question, "What's in it for me?" In particular, motivation must answer these two questions.

1. How will my investment in a software assessment process, as part of a process improvement program, help me improve my abilities as a manager to reach the goals of my organization?

2. How will my participation in a software assessment process help me produce software with less effort and fewer problems?

Management will benefit through improved customer satisfaction, reduced cost for value, increased timeliness, and improved operational quality. Improved software processes result in improved market performance, due to increased software reliability and functionality.

The software practitioner is motivated through realization of a consistent software development/maintenance environment that has an established and well documented process. Additional benefits to the software engineer are the amount of reduced rework and the satisfaction of producing a quality product. There is more time for innovative work since less time is spent on rework.

3.3. Quality Benefits

The software products developed in a mature environment include fewer software defects, resulting in less system down time and improved operational results for the customer. Additional quality benefits are more customer/operator satisfaction, less software maintenance, reproducible and known software configuration control, and a high confidence level in the resulting software product.

3.4. Resource Allocation/Schedule Estimation

As the software process becomes repeatable (CMM Level 2) and defined (CMM Level 3), and in general independent of individuals engaged in heroic efforts, the necessary personnel resources can be assigned and the completion dates forecast with known reasonable accuracy. With this software process in place, successful software projects are not dependent on individuals, and software developers can be more easily exchanged with minimal project impact.

3.5. Compliance Benefits

Documentation obtained from a software process assessment will help to satisfy a DOE or internal audit. The following chart cross-references the SEI Capability Maturity Model's eighteen Key Process Areas to several different requirements documents. The numbers in the cells indicate that the Key Process Area corresponds to that paragraph number in the indicated requirements document. Definitions for the Key Process Areas can be found in Appendix G. Full references and titles to the requirement documents can be found in Appendix H.

Key Process Areas	QC-1 Rev. 7.0 4/20/93	QC-2 2/22/90	5700.6C Criteria 8/21/91	1330.1D 5/18/92	NQA-2 part 2.7 5/31/90	ISO 9001 1994
1. Requirements management		(2.1)	6	Att. 1I2b	3.1, 6.2, 7.1	4.3
2. Software project planning		(2.2) (4.2)	1	Att. 1I1a, b, c, d	3.	4.1, 4.2, 4.3, 4.9
3. Software project tracking/ oversight		(2.3)	1, 6	Att. 1I1a, d		4.1
4. Software subcontract management	(5.1)		7	8o, Att. 1I1d	10.	4.3, 4.6, 4.7
5. Software quality assurance	15.0	(1.0)	3,10, 6, 8, 9	8d, Att. 1I1e, f, Att. 1I2f, i, l	(all); 6.1	4.1, 4.2, 4.9, 4.17
6. Software configuration management	15.0	(3.1)	4, 6, 8	Att. 1I3b	5.	4.4, 4.5, 4.8, 4.12, 4.13, 4.14, 4.15, 4.16
7. Organization process focus	15.0		5	8c, 10, Att. 1I2a		
8. Organization process definition	15.0		5	8e, 8h, 10, Att. 1I2a		4.20
9. Training program	(3.1)	(3.4)	2	Att. 1I2k		4.18
10. Integrated software management			(all); 1	8b, 8e, Att. 1I1a, b, Att. 1I2k		4.7
11. Software product engineering		(3.2) (4.1)	6	8e, 8n, Att. 1I2b, e, f, g, h, j, m, n, Att. 1I3a, c		4.2, 4.4, 4.5, 4.8, 4.9, 4.10, 4.11, 4.12, 4.15, 4.16
12. Intergroup coordination			1	8b, 8i, 8j, 8k, 8l, 8m, 9, Att. 1I1g, Att. 1I2c		
Key Process Areas	QC-1 Rev. 7.0	QC-2	5700.6C Criteria	1330.1D	NQA-2 part 2.7	ISO9001 1994

	4/20/93	2/22/90	8/21/91	5/18/92	5/31/90	
13. Peer reviews		(4.3)	4, 6	Att. 1I1f, Att.1I2i		4.1, 4.4, 4.10, 4.16
14. Quantitative process management		(2.3)	3	8d, Att. 1I1e		4.20
15. Software quality management			3, 9, 10	8d, Att. 1I1e, f, Att. 1I2d, i, l	6.1	4.20
16. Defect prevention	(3.3)	(4.3)	3, 8			4.14
17. Technology change management			10, 6	8e, 8i, Att. 1I1d		
18. Process change management			3, 5			

Note: Parenthetical references, e.g. (4.3), address the Key Process subject, but do not specifically address software.

This page is left intentionally blank.

4. Assessment Alternatives Summaries

The following sections: provide an overview of the SEI, identify the preliminary planning that needs to be done before conducting a software process assessment, outline the two SEI based assessment methods, and discuss the follow-on activities needed to implement software process improvement.

4.1. SEI Overview

The Software Engineering Institute (SEI) was established in 1984 at Carnegie-Mellon University (Pittsburgh). It resulted from a Department of Defense initiative and its purpose is *to promote and advance various areas of Software Engineering*.

One output from the SEI was the creation of the Capability Maturity Model (CMM), the maturity questionnaire, and the assessment process. The CMM and the maturity questionnaire are the foundation of the assessment. The CMM represents a framework that helps organizations improve their software. Using the CMM and the maturity questionnaire, the assessment process measures organizations against that framework to determine the maturity of the organization and to identify the major improvement areas. More information on SEI can be found in Appendix G.

4.2. Preliminary Planning

To achieve a successful assessment some preliminary planning needs to be done. The following list identifies the activities that will help establish the necessary foundation for either type of assessment. It is essential that management "buy-in" be obtained before attempting either assessment. If it does not occur, the chances of conducting a successful assessment, and obtaining resources for follow-on improvement initiatives, are slim.

Plan:

1. Present to management an overview of: the SEI, an assessment, and the rest of the assessment implementation plan. This helps to gain sponsorship and resources for an assessment.
2. Determine which assessment method to implement.
3. Create a Software Engineering Process Group (SEPG).
4. Introduce the assessment process to the organization.

The above plan will put in place the resources and management support needed to conduct an assessment. Its strengths lie in obtaining management and staff support from the start. It relies heavily on up front preparation and knowledge. The goal is to have the infrastructure in place that is capable and has the responsibility to improve the software process before an assessment is attempted. This up front effort improves the odds that the whole assessment and improvement processes will be successful.

Staffing Requirements:

Below is a summary of the activities, roles, tasks and staffing requirements for the planning activities. A more detailed breakdown of these can be found in Appendix A.

The staffing requirements identify the total effort needed to complete the specified task(s). Depending on the task, that effort could be one activity completed in a single day or multiple activities completed across many days. Remember these are only guidelines. Any of the items can change depending on the organization and the staff available.

It is recommended that a charge number be assigned to better track the actual time commitments.

Activity	Role/Task	Staffing Requirements
Step 1 - Present an overview	Assessment Champion* - Preparation/Present	1 day
	Management - Presentation	1-2 hours
Step 2 - Determine assessment method	Assessment Champion*, Management - Make decision	2 hours
Step 3 - Create SEPG	Assessment Champion* - Provide Guidance	1 day
	Management/Staff - Create SEPG	1 day
Step 4 - Introduce assessment	Assessment Champion* - Preparation/Present	1 day

*The Assessment Champion refers to the individual leading the assessment effort. This could be the site NWC Software Quality Assurance Subcommittee (SQAS) representative, a project leader, a supervisor, manager, etc. This person must understand the benefits of an assessment and be willing to sell those benefits to management and staff.

4.3. Assessments

This section introduces two different assessment methods:

1. A self-assessment is conducted by software organizations or software development teams and
2. A formal assessment conducted by experienced SEI assessors (such as a licensed SEI vendor).

The self-assessment is an informal and less expensive implementation of the formal assessment. It provides many of the same benefits but lacks much of the formality of the formal assessment. Organizations who do not have the resources for a full formal assessment should evaluate each step of the formal assessment plan and determine if a less costly version of the formal assessment can be implemented. The goal should be to develop an assessment plan that the organization can and has the resources to implement.

Both methods have the same objectives:

1. Identify the maturity of the assessed organization's software process and
2. Just as important, identify the Key Process Areas that need improvement and create the beginning of an improvement strategy.

An outline of both assessment methods follows, more details can be found in the Appendices B and C.

4.3.1. Self-Assessment

Certainly in order to make significant strides in software process improvement it is essential to have management commitment. But there are some things a software development organization can do without a strong commitment from management or a significant drain of resources from project activities.

This section describes the steps that a software development team might take to assess their software processes using minimal resources.

Plan:

1. Establish the assessment team (or use SEPG).
2. Review the Capability Maturity Model (CMM).
3. Evaluate projects using the questionnaire.

Staffing Requirements:

Again, the staffing requirements identifies the total effort needed to complete the specified task(s). Depending on the task, that effort could be one activity completed in a single day or multiple activities completed across many days. Remember these are only guidelines. Any of the items can change depending on the organization and the staff available. Appendix B provides more details.

Activity	Role/Task	Staffing Requirements

Step 1 - Determine assessment team	Assessment Champion/SEPG/Management - Determine team	2 hours
Step 2 - Review CMM	Assessment Team - Review CMM	3-5 days
Step 3 - Evaluate projects	Assessment Team - Evaluate projects	3-5 days
	Staff - Answer questions	4 hours

4.3.2. Formal Assessment

The formal assessment is a well documented process. It is led by individuals who have experience in conducting an SEI assessment. The time and resource commitments are considerable, but the results will provide a clear path to software process improvement. The assumption is that the management decision and budgetary activities have been completed during Preliminary Planning (Ref. 4.2) and that a strong commitment has been made.

Plan:

1. Determine an Assessment Team and Leader.
2. Conduct a formal assessment.
3. Prepare a detailed written report of the assessment results and recommendations.

Staffing Requirements:

Activity	Role/Task	Staffing Requirements
Step 1 - Determine Assessment Team Leader	Assessment Champion, SEPG, Management - Determine leader	5 days (Depends on if a licensed vendor is being used)
Step 2 - Conduct formal assessment Phase 1. Planning Phase 2. Training Phase 3. Assessment Phase 4. Final Findings Presentation	Assessment Champion, Assessment Team Leader, SEPG, Management - Determine coordinator	4 hours

Phase 5. Executive Session		
	Assessment Team Leader - Lead assessment	28 days
	Assessment Team Coordinator - Participate in assessment	21 days
	Staff - Assessment Team Member (2-6) - Participate	13 days/Assessment Team Member
	Staff - Project Leader (max 4) - Answer questions	1 day/ Project Leader
	Staff - Functional Area Representatives (20-30% of organization) - Answer questions	1 day/Functional Area Representative
	Senior Management - Provide support	1 day
	Middle Management - Provide support	2 hours
	Administrative Support - Provide administrative support	2 days
Step 3 - Report results	Assessment Team - Prepare / Present	As Needed

The staffing requirements represent the total effort. Depending on the task, the staffing requirements may cover one continuous time period or multiple time periods across days or weeks. Remember these are only guidelines. Appendix C provides more details.

4.4. Improvement Activities

When either assessment is complete, follow on activities are needed to start the improvement process.

Plan:

1. Develop improvement action plan.
2. Implement the plan.

Staffing Requirements:

Activity	Role/Task	Staffing Requirements

Step 1- Develop improvement action plan	Assessment Team/SEPG - Develop plan	As needed
Step 2 - Implement the plan	SEPG - Implement plan	As needed

The staffing requirements depend upon the size of the improvement effort. The organization level of effort must be balanced against the availability of staff. The goal is to form a plan and make progress towards implementing that plan. Appendix D provides additional details.

This page is left intentionally blank.

5. Summary

An assessment program requires significant time and commitment by an organization. To achieve the full benefits of an assessment, time and commitment must be sustained. Like any improvement program, long term commitment and support is required to achieve its full benefits.

The purpose of the assessment is to provide a start for software process improvement. Once most of the assessment recommendations are implemented, the assessment process should be repeated. An organization should repeat the entire assessment process every 18 months to two years. This provides a measure of how well the organization is doing towards improving their software processes.

The appendices that follow detail the assessment process, the time commitment, roles and responsibilities, and other information determined to be useful in understanding the assessment process. They are:

A. Assessment - Preliminary Planning - Discusses the preliminary planning activities in detail. Also included is a readiness survey.

B. Assessment - Self-Assessment - Discusses the self-assessment activities in depth. It includes a sample assessment form.

C. Assessment - Formal Assessment - This appendix covers the formal assessment and provides details for each step.

D. Assessment - Improvement Activities - This covers the beginning of the improvement process and how the assessment results can help.

E. Software Engineering Process Group (SEPG) - This appendix discusses the SEPG and how to establish it. It also addresses the commitment needed to make the SEPG successful.

F. Guidelines for Selecting Assessment Team Members and Project Selection - The purpose of this section is to provide suggestions on selecting assessment team members. It includes details on roles and responsibilities of each member for a formal assessment but can also be applied to a self-assessment.

G. Software Engineering Process Maturity Levels, SEI Questions and Key Process Areas - This section provides some background on the SEI, the CMM, Key Process Areas, and identifies the questions that might be used during an assessment.

H. References - Additional references to assist the reader.

This page is left intentionally blank.

A. Assessment - Preliminary Planning

The goal of this section is to establish the foundation to achieve a successful assessment. The amount of effort required for these activities will vary depending upon the initial staffing levels and available time. If the need for an assessment is driven by management, then the steps defined here can confirm and increase management support. If the need for an assessment is internally driven by the organization's staff, then these steps may be less important. It's up to each organization to review each step and implement appropriately.

At the end of this section is a readiness survey which should help determine if an organization is ready for an assessment.

A.1. Plan

This section will expand and explain each step of the preliminary planning. Its goal is to help build the framework needed to conduct a successful assessment.

Step 1. Present an overview of the SEI, an assessment, and the rest of the assessment implementation plan to management to gain sponsorship and resources for an assessment.

An introduction to software process improvement and the assessment process can help obtain the sponsorship needed to conduct a successful assessment program. One of the major keys to achieving a successful assessment is the commitment by the organization's management to the actions required to achieve software process improvement. Quoting **pragma Systems** (an SEI licensed vendor) in reference to a formal assessment:

> "Without that commitment, the assessment **will not** succeed because the follow-up actions will not be supported. In fact, the organization will be worse off than if it had not conducted an assessment at all. At the closing of an assessment, the technical professionals see that the state of the organization has been clearly presented to management. If management does not respond with improvement actions, the technical professionals will be demoralized, thinking that their managers are either incompetent or uncaring. Therefore, it is essential that the senior manager recognize the raised expectations of the technical professionals following an assessment. The opportunity to demonstrate real commitment comes after the assessment."

To complete this step an overview will need to be created. It will explain the SEI, the assessment program, software process improvement and the benefits, including the role of an assessment in forming a cost effective, successful plan for the intentional

improvement of the organization's software process. It will also include the time commitment required to conduct the assessment and the time which will be required to implement the improvement recommendations.

Step 2. Determine which assessment method to implement.

Two assessment methods are presented in this document. Both are based upon the SEI Capability Maturity Model and the SEI assessment process. Both also have advantages and disadvantages. The team determining which assessment process to use should be familiar with both. Depending on the organization's goals and available resources one, both, or parts of both assessment processes may be implemented.

The formal assessment has the advantage of being a well defined process. It provides an organization with both an indication as to its software process capability and the basis for a software process improvement plan. A formal assessment needs to be led by experienced personnel in order to achieve success. SEI licensed vendors are available to lead an SEI assessment. They provide the leader and 1-2 additional team members. The only drawback is the cost associated with contracting with a vendor. The biggest advantage is the industry accepted SEI assessment process and the credibility that a vendor provides. Also, any follow-on assessment may be led by individuals who are part of the assessment team.

The self-assessment can provide the same advantages but lacks some of the credibility. If management commitment is not strong then a self-assessment or a scaled-down version of the formal assessment should be considered because they will use less resources. The self-assessment's biggest drawback is that it relies heavily on the knowledge and experience of the assessment team. Its biggest advantages are that the costs are lower and it can be spread across a longer time period.

Determining which assessment to use requires knowledge of both methodologies. Staffing, scheduling and money may play an important role in the decision. The goal is to implement some type of assessment methodology so that software process improvement has some guidance as to which goals to obtain first.

Step 3. Create a Software Engineering Process Group (SEPG).

This step will be the first formal step taken towards improving an organization's software process. It is an important step because management must commit time and resources. The Software Engineering Process Group (SEPG) is the focal point and central force for process improvement in the organization. The group maintains an overall view of current processes and facilitates the use of new and improved processes.

The SEPG is staffed with software engineers with line experience. Its activities include:

- Obtaining and maintaining management support,

- Facilitating software process assessments,
- Planning improvement approaches and their implementation activities,
- Working and consulting with projects to implement process improvements,
- Arranging training or continuing education,
- Creating and maintaining process definitions in conjunction with management and staff, and
- Maintaining the site process library.

The establishment of this group is critical to the success of the process improvement. The individuals named to the group must understand the current way the organization develops software, be given the task as a new project, and be given the time and the budget to work on it.

The SEPG should establish contact with other NWC or industry Software Engineering Process Groups. Other SEPG's efforts should be tracked and monitored to avoid duplication of work. Participation with other SEPGs is possible with support from management.

Appendix E provides information on the formation of an SEPG and the type of individuals who should participate.

Step 4. Introduce the assessment process to the organization.

Another key to a successful assessment is a desire by the organization to improve and allow the best and brightest people to get involved in that improvement process. The purpose of this step is to explain the assessment to the individuals who will be directly involved and to gain support for the assessment. This presentation may also discuss the roles and tasks which individuals may do during the assessment. This step will involve introducing SEI, the assessment process, and the assessment implementation strategy, potential benefits, their role and any organization issues which need to be addressed.

To complete this step, an overview will need to be created. It should be similar to the one done for management. If less formality is needed, this step can be accomplished via one-on-one meetings with the team members.

Self-Assessment Readiness Survey

Organization: _____ Date: _____

Directions: Complete this questionnaire for each organization you plan to assess before you commit to the assessment. The answers to the questions will provide you a general indication of the areas that will need attention before the assessment and after the assessment to facilitate process improvement activities. Keep in mind that this instrument is designed to provide a global indication and does not represent all the details of the issues you will face in your process improvement work.

For each question below, circle the appropriate number to rate the extent to which the statement is true. If you don't know the answer to a question, leave it blank.

Key:

1. Hardly at All
2. To a Mild Extent
3. To a Moderate Extent
4. To a Great Extent

	Hardley at All	Mild Extent	Moderate Extent	Great Extent
Sponsorship				
The sponsor (*the senior manager who has the position and clout to initiate and sustain the assessment and process improvement activities*) is dissatisfied with the current state.	1	2	3	4
The sponsor is able and willing to demonstrate the type of public support necessary to convey strong organizational commitment to the assessment.	1	2	3	4
The sponsor is willing to commit resources to do the assessment and the follow–up improvement activities.	1	2	3	4
The sponsor has the power and resources to support improvement plans.	1	2	3	4
The sponsor is willing to assure that progress and problems will be tracked during follow–up improvement.	1	2	3	4
The sponsor is aware of the personal, organizational, and political cost of the change coming from the assessments.	1	2	3	4
Sponsorship Total:	_____			

	Hardley at All	Mild Extent	Moderate Extent	Great Extent
Culture				
There is consistency between our organization's view of teamwork (*how much people are encouraged to communicate and cooperate with each other*) and that established during a process assessment (*e.g., collaboration, consensus decisions, etc.*).	1	2	3	4
There is consistency between the way goals, tasks, and role assignments are currently defined and the way they're expected to be defined when beginning a process improvement effort.	1	2	3	4
Culture Total:	_____			
Resistance				
Practitioners see a need for process improvement.	1	2	3	4
Managers see a need for process improvement.	1	2	3	4
The communication between management and practitioners is clear and direct, with little confusion.	1	2	3	4
The cost of process improvement in time and personnel is seen by management as reasonable.	1	2	3	4
Practitioners are confident that management will provide organizational support (time, money, personnel) to help improve the software process.	1	2	3	4
The anticipated impact on budgets and schedules is seen by management as a reasonable cost of process improvement.	1	2	3	4
The organization has a history of success in making changes.	1	2	3	4
The person sponsoring the assessment inspires confidence in his or her subordinates.	1	2	3	4
The assessment team members are widely trusted and viewed as effective in their work.	1	2	3	4
Past performance in software improvement is viewed by sponsors and managers as a springboard to improvement.	1	2	3	4
Practitioners and project managers believe that the sources for information gathered in the assessment will remain confidential.	1	2	3	4
Resistance Total:	_____			

	Hardley at All	Mild Extent	Moderate Extent	Great Extent
Synergy				
Groups in the organization communicate with each other directly and with few misunderstandings.	1	2	3	4
Managers tend to create an open atmosphere where differences in opinion can be surfaced and dealt with directly.	1	2	3	4
Task groups generate creative ways to merge their diverse perspectives into alternatives everyone supports.	1	2	3	4
Synergy Total:	_____			
Organizational Issues				
The stress from our day–to–day workload is low enough for us to manage the changes that come as a result of the process assessment.	1	2	3	4
Our organization has clear lines of responsibility and authority which creates a tendency to get results rather than protect turf.	1	2	3	4
Our organization provides our employees the latitude to make mistakes, which encourages a risk–taking environment.	1	2	3	4
When changes are introduced, there are negative consequences for failing to support the changes.	1	2	3	4
When changes are introduced, employees are rewarded for supporting the changes.	1	2	3	4
When management announces a strategic directive, the organization always acts on it no matter how many other projects or problems compete for resources.	1	2	3	4
Decision–making for a major change effort involves obtaining true consensus from the people involved, rather than just the appearance of consensus.	1	2	3	4
Managers don't resist making changes when turf or control is at stake, because the corporate goal always has priority over the goals of individual groups.	1	2	3	4
Organizational Issues Total:	_____			

Summary of Responses

Directions: List your total scores from each category in the questionnaire next to the category on this sheet. Mark an X or a S on the scale for each category to determine your potential risk in conducting an assessment or for subsequent process improvement. You can then connect your marks to get a profile of your current readiness.

		Doubtful				**Caution**				**Favorable**			
	Totals	3	6	9		12		15		18	21		24
Sponsorship	_____												
		1	2	3		4		5		6	7		8
Culture	_____												
		6	11	17		22		28		33	39		44
Resistance	_____												
		1	2	3	4	5	6	7	8	9	10	11	12
Synergy	_____												
		4	8	12		16	20	24		28	32		
Organizational Issues	_____												

Interpretation Guidelines

The first two categories – Sponsorship and Culture – are critical for conducting as assessment. Your total score for each category should be in the "Favorable" range for you to anticipate success in conducting a useful assessment.

The last three categories – Resistance, Synergy, and Organizational Issues – are critical for initiating process improvement activity. The ideal state for an effective start at process improvement would have the scores for these categories in the "Favorable" range.

In any category, those scores not in the favorable range indicate that you must do some groundwork for the organization's efforts in assessment or process improvement to be successful. Review the questions that have been marked with a "1" or a "2" or questions you didn't answer for areas that must be addressed.

This page is left intentionally blank.

B. Assessment - Self-Assessment

B.1. Roles and Time Commitment

This section describes the steps that an organization or a software development team might take to improve their software processes using minimal resources. The self-assessment requires less management commitment and resources. Its only staffing requirements are the SEPG/Assessment Team, individual project leaders and software development staff. The time commitment varies depending on the amount of time and resources available.

B.2. Plan

Step 1. Establish the assessment team (or use SEPG)

Establish a team of individuals to serve a function similar to the SEPG or use the SEPG already established. They must be willing to take responsibility for the development and management of software processes for the organization. This team represents a cross section of the types of projects within the organization and if possible, should also represent various software engineering disciplines, i.e. software requirements analysis, software design, coding, software test, software configuration management, and software quality assurance.

Step 2. Review the Capability Maturity Model (CMM)

Review the Capability Maturity Model (CMM) for Software (see SEI Technical Report: *Capability Maturity Model for Software Version 1.1)* since this model provides a foundation for the development of methods which are useful in software process improvement. This document provides an understanding of the Key Process Areas that are part of effective processes for developing or maintaining software and identifies the items needed to achieve the next maturity level in the CMM

Step 3. Evaluate projects using the questionnaire

Using the SEI Questionnaire, select a level you think is beyond your current maturity level but within reach. Extract the questions for that level from the questionnaire and evaluate several projects in the organization against those questions. You may need to define some of the terms in the questions to relate to your specific environment. For example, “development manager” on the questionnaire might equate to “software project lead” in your organization. This assessment will serve as a baseline for the organization.

B.3. Sample Assessment Form

Below is an example assessment. See Appendix G for additional information on the SEI questionnaire.

Sample Assessment
Using 1987 SEI Level 2 Questions

199? SEI Self-Assessment of Department Software Engineering Capability					
No.	**SEI Question -Level 2**	**Project 1**	**Project 2**	**Project 3**	**Notes**
1.1.1	For each project involving software development, is there a designated software manager?	Y	Y	Y	sw mgr=sw project lead
1.1.2	Does the project software manager report directly to the project (or project development) manager?				project sw mgr =sw project lead project mgr=system project lead
1.1.3	Does the Software Quality Assurance (SQA) function have a management reporting channel separate from the software development project management?	Y	Y	Y	
1.1.6	Is there a software configuration control function for each project that involves software development?	Y	Y	Y	
1.2.2	Is there a required training program for all newly appointed development managers designed to familiarize them with software project management?				development mgr = sw project lead
1.3.1	Is a *mechanism* used for maintaining awareness of the state–of–the–art in software engineering technology?				tools advisory team, Technical User Group meetings
2.1.3	Is a *formal procedure* used in the management review of each software development prior to making contractual commitments?	?	Y	Y	Engineering Procedures Manuals
2.1.4	Is a *formal procedure* used to assure periodic management review of the status of each software development project?				mgmt review = status meetings; not formal procedure
2.1.5	Is there a *mechanism* for assuring that software subcontractors, if any, follow a disciplined software development *process*?				
2.1.7	For each project, are independent audits conducted for each step of the software development *process*?	Y	Y	Y	QA activities
2.1.9	Are coding *standards* applied to each software development project?	Y	Y	Y	
2.1.14	Is a *formal procedure* used to make estimates of software size?				
2.1.15	Is a *formal procedure* used to				

	produce software development schedules?				
2.1.16	Are *formal procedures* applied to estimating software development schedules?				
2.1.17	Is a *mechanism* used for ensuring that the software design teams understand each software requirement?				mechanism = inspections design team not always included
2.2.1	Are software staffing profiles maintained of actual staffing versus planned staffing?				
*2.2.2	Are profiles of software size maintained for each software configuration item, over time?				
*2.2.4	Are statistics on software code and test errors gathered?	Y	Y	Y	to some extent
2.2.7	Are profiles maintained of actual versus planned software units designed, over time?				
2.2.8	Are profiles maintained of actual versus planned software units completing unit testing, over time?				
2.2.9	Are profiles maintained of actual versus planned software units integrated, over time?				
2.2.10	Are target computer memory utilization estimates and actuals tracked?				low priority; may not apply
2.2.11	Are target computer throughput utilization estimates and actuals tracked?				low priority; may not apply
2.2.12	Is target computer I/O channel utilization tracked?				low priority; may not apply
2.2.16	Are software trouble reports resulting from testing tracked to closure?	Y	Y	Y	
2.2.18	Is test progress tracked by deliverable software component and compared to the plan?	Y	Y	Y	
2.2.19	Are profiles maintained of software build/release content versus time?				
*2.4.1	Does senior management have a *mechanism* for the regular review of the status of software development projects?				senior mgr = dept. mgr and director
2.4.5	Is a *mechanism* used for regular technical interchanges with the customer?	Y	Y	Y	Technical Interchange Meetings
*2.4.7	Do software development first–line managers sign off on their schedules and cost estimates?				first-line mgr = dept. mgr

*2.4.9	Is a *mechanism* used for controlling changes to the software requirements?		Y		Requirements Tracking System
*2.4.17	Is a *mechanism* used for controlling changes to the code? (Who can make changes and under which circumstances?)	Y	Y	Y	configuration control
2.4.20	Is there a *mechanism* for assuring that regression testing is routinely performed?	Y	Y	Y	

C. Assessment - Formal Assessment

C.1. Roles and Time Commitments

The formal assessment requires the following roles to be filled: Assessment Team Leader, Assessment Team Coordinator, Assessment Team Members, Project Leaders, and Functional Area Representatives. Additionally Senior and Middle Management and Administrative Support personnel also have roles in the assessment process. The roles and time commitment have been fully outlined in Appendix F. It addresses the following issues:

- Assessment participants and supporting personnel,
- Roles and responsibilities for Assessment Team Members, and
- Selection of projects and other assessment personnel

C.2. Plan

Step 1. Determine an Assessment Team Leader.

The purpose of this phase is to identify who will lead the assessment. A knowledgeable experienced assessment team leader is needed to conduct a successful assessment. The leader must be knowledgeable enough to train the team in the process and the assessment techniques.

Leaders can come from multiple sources:

SEI - Normally, SEI does not conduct assessments. In unique circumstances they will.

SEI licensed vendors - One of the most effective ways to do an assessment is to be advised by professionals from an SEI-licensed vendor. At this time there are nine vendors licensed by SEI to conduct assessments. Using a licensed vendor also lends the assessment and the process additional credibility.

Other organizations - The third choice is to find an individual from another source. That individual must have detailed knowledge of the SEI assessment and experience at leading an assessment.

The decision to hire a vendor may seem to be expensive, but to cost effectively conduct an assessment you must have an assessment leader experienced in conducting assessments and who can train and lead the team through an assessment. The assessment process is a very structured process. When managed properly, an appraisal of an organization's current software process will be completed and the organization can go

forward with the suggestions made. When done incorrectly, the assessment can cause major damage to the organization through decreased morale and wasted effort.

Pick a leader who can relate to your site and if possible, has experience in the areas you hope to assess. Someone with assessment experience has the best chance of leading the assessment team to success.

Step 2. Conduct a formal assessment

Assessment Schedule

Planning by the Assessment Team Leader and Assessment Team Coordinator should begin 4 - 6 weeks before the on-site period.

Phase	What	When
Planning	Identify all participants, select projects, handle logistics, schedule briefings, inform the organization, schedule Assessment Team Member meetings	Begin 4 - 6 weeks before the assessment
Training (4-5 days)	Team building, organization overview, assessment walkthrough, Executive Briefing, participant briefing, SEI Questionnaire, response analysis.	Approx. 3 weeks before the assessment
Assessment (4 days)	Opening meeting, conduct interviews and discussions, prepare preliminary findings, dry-run with participants to produce final findings.	The assessment
Final Findings Presentation (1/2 day)	Present to Senior Management, conduct individual management meetings, debrief assessment team.	Last day of the assessment
Executive Session (1/2 day)	Private meeting with the Senior Management to explain the findings in detail and to answer questions.	Last day of the assessment

Re-assess the organization approximately 18 - 24 months following this assessment.

Phase 1. Planning

This phase establishes the assessment foundation. The Assessment Team Leader will have been identified. Once that is completed the Assessment Team Leader will work with the organization's management to determine the Assessment Team Coordinator. The Assessment Team Coordinator is the person on site who is responsible for making sure the assessment happens. The Assessment Team Leader will then train and help the Assessment Team Coordinator to:

- Define the Assessment Team Composition
- Define the scope of the assessment
- Select the Assessment Team
- Select the Projects to be assessed
- Determine the Functional Area Representatives

- Select the Project members to participate in the Assessment
- Complete the logistics for the Training Phase
- Complete the logistics for the Assessment Phase
- Coordinate the Assessment Process

This phase will be done via individual phone conversations, conference calls, and/or meetings across a 4-6 week period. The Assessment Team Leader will provide briefing materials, project work sheets, and detailed checklists. The Assessment Team Leader will talk with Senior Management to explain their role and responsibilities.

Phase 2. Training

Once all the logistics are complete the training week (4-5 days) can begin at the site. The training week consists of three pieces: orientation, team building, and assessment set-up. The team is trained on:

- Principles of Software Process Management and how they relate to Total Quality Principles,
- Team Building,
- Mechanics of an SEI Assessment, and
- SEI Questionnaire, Capability Maturity Model, Key Process Areas

Project teams are trained in the Capability Maturity Model, Key Process Areas and the Assessment Process. Additionally, the Assessment Team Coordinator gives an organizational overview briefing so that all Assessment Team Members have the same view of the organization.

The Assessment Team Leader gives an executive briefing about the assessment process to senior management (i.e. site president and/or manager of the assessed organization). They are asked to make the assessment and the resulting software process improvement efforts visible through memos, announcements, local newsletters, and management meetings.

The second part of the training week consists of getting ready for the assessment. The following activities are accomplished:

- Review the projects to be assessed and finalize the selection,
- Determine the current life cycle phases of the organization's software development life cycle,
- Determine who the Assessment Team will interview,
- Determine the topics for the Function Area Representatives groups,

- Determine which and how many technical professionals from the organization will be involved with the Function Area Representatives group discussions,
- Determine all the roles and responsibilities for each Assessment Team Member for each step of the assessment,
- Review the schedule for the Assessment Participants Briefing, and
- Review the draft schedule for each hour of the on-site assessment period.

The third and last section of the training week consists of meetings with everyone involved in the assessment process. This includes not only the assessment team but the Project Leaders and the Functional Area Representatives who will be interviewed during the assessment week. All these individuals are brought together for the Assessment Participants Briefing. This briefing describes their role in the process, thanks them for their help and provides them with the initial schedule and when they will be needed.

Upon completion of this briefing the project leaders are asked to complete the SEI Questionnaire. The responses are then summarized and the Preliminary Response Analysis is started. Response Analysis is the review of the projects' SEI Questionnaire responses. The review is a search for inconsistencies within and across projects. These inconsistencies serve as a starting point for creating the questions which will be asked of the projects during the assessment phase.

The results of the questionnaire are also compiled into the organization's preliminary maturity level. The final level is not determined until the end of the assessment.

Phase 3. Assessment

The assessment at the site will follow the training by 2-3 weeks. During that time the final logistics are worked out. The format and schedule of the assessment week is firmly set. The following is an outline of the activities.

- Brief opening supportive comments from the site president or manager of the assessed organization.
- Finalize Response Analysis (started during the training week).
- Interview the Project Leaders (3-4).
- Hold the Functional Area Representatives group discussions.
- Review the results of the interviews and discussions.
- Identify the Preliminary Results.
- Review the Preliminary Results with the Project Leaders to gain consensus.
- Create the Final Results.

- Present the Final Results to the Functional Area Representatives to gain agreement and make sure there are no misunderstandings.
- Present the Final Results to the Project Leaders to gain agreement and make sure there are no misunderstandings.
- Formalize the Final Results.

Phase 4. Final Findings Presentation

This phase is short and consists of the following activities.

- Present the Final Results to everyone involved including senior management.
- Senior management gives closing comments.

For most of the participants, the assessment is officially completed after the presentation.

Phase 5. Executive Session

Once the formal assessment itself is completed, the Assessment Team Leader and the Assessment Team Coordinator meet with senior management to review the results in depth and discuss the next steps. It is mandatory that senior management attend. If they do not, the meeting is rescheduled. This phase provides a private session allowing the Assessment Team Leader and Assessment Team Coordinator to answer questions and provide their private input.

Step 3. Prepare a detailed written report of the assessment results and recommendations.

The Assessment Team Leader and the Assessment Team Coordinator meet with the Assessment Team one last time. During that meeting the Assessment Team discusses any recommendations which came out during the assessments and the next steps. A formal written assessment report follows in 2-3 weeks.

This page is left intentionally blank.

D. Assessment - Improvement Activities

D.1. Roles and Time Commitment

Software process improvement activities should include everyone involved in the software process. This includes programmers, testers, software quality assurance, designers, etc. The time commitment varies depending on the amount of time and resources available. Appendix E addresses some of the roles and time commitments as well as the type of individuals who should be considered for the SEPG and the software process improvement team.

D.2. Plan

Step 1. Develop improvement plan

The formal assessment identifies major issues which need to be addressed to improve the organization's software process. If a Software Engineering Group has not been formed it should be at this time. The organization's management, the assessment team coordinator, and the SEPG meet to discuss the assessment results and recommendations. They take the assessment results and the recommendations and create an improvement plan which identifies and prioritized the improvement activities. That plan should then be presented to management for their final approval and the improvement process begins.

For a self-assessment, the team may create the improvement plan from the questions. Select no more than ten of the questions that the organization could realize a benefit from improving. From this list of ten, place them in order of priority for the organization. Brainstorm about each question and select an action that addresses the issues covered in the question. Develop an action plan for how to improve in this area that is within your circle of influence. If management is involved, present the plan to them to obtain support.

Step 2. Implement plan

To implement the plan, try to set up periodic meetings of the SEPG and work on one area at a time. This is a long term commitment to process improvement. Many times it is possible to adapt or embrace methods which have been developed by other organizations in the corporation or even by other corporations or NWC sites.

Perform a reassessment on a periodic basis depending on the length of the software projects within the organization and try to get management commitment and resources dedicated to software process improvement.

This page is left intentionally blank.

E. Software Engineering Process Group (SEPG)

Software process improvement requires four groups of participants:

1. Top management who must initiate and visibly support software process improvement.
2. A Management Steering Council, composed of senior management, who provide sponsorship, management oversight, support for SEPG activities and resources, and clear a path within the site. Senior management is defined to be those managers who make the day-to-day operational decisions.
3. A Software Engineering Process Group (SEPG), composed of technical staff members, who do and direct the process improvement work.
4. Improvement Teams, composed of technical staff members or management, who work on individual process improvement problems and projects. These teams are created by the SEPG and the Management Steering Council.

Two of these groups, the Management Steering Council and the SEPG should be permanent and assigned the task, by top management, to direct the process improvement activities.

The objective of the SEPG is to improve the quality and productivity of the software activities. The SEPG is the focal group for action planning, process improvement, technology implementation, etc. They are also responsible for the exchange of information between improvement efforts within the site, industry, and the NWC.

See the SEI paper SEI-90-TR-24 Software Engineering Process Group Guide for additional help and guidance.

E.1. Overview

Strategy

Process improvement needs are not static; they change and evolve. It is important that those responsible for guiding process improvement efforts recognize this and constantly re-evaluate and update the organization's needs.

The organizations have varied responsibilities, and therefore their needs vary. It is important that the framework and support for the improvements are provided - including the proper guidance, visibility, and resources.

If the process improvement efforts are to become a part of daily business, then they need to be continuous and repeatable. Meetings must be regular, with set agendas and published minutes. Action items must be tracked to closure. By assigning action items Risks To SEPG Success and tracking them to closure, the organization is able to ensure

problems aren't allowed to fall through the cracks and not get addressed. Quite frequently, the items that are most difficult to address will provide the most benefit. Total Quality training provides an excellent framework for the operation of this group.

Risks To SEPG Success

The lack of senior management support of this improvement team may result in:

- The SEPG not having enough full-time capable professionals to do competent work.
- The SEPG not having sufficient management support to convince the projects to support the process improvement efforts.
- The SEPG leader not being able to obtain the participation of the most knowledgeable software professionals in the process improvement task groups.

Studies have shown and the Software Engineering Institute has stated that without good support the SEPG effort will quickly die and an organization could be worst off than they were before.

Financial Commitment

The Software Engineering Institute's recommended budget for an SEPG is 1% of the total budget of the software organization it supports.

Recommended budget for software process improvement is an additional 1% to 3% of the total budget of the organization it supports.

E.2. SEPG Setup

The questions to address in setting up an SEPG are:

- How much staffing is required?

  A useful guide is to aim at full-time assignments to the SEPG of about 2% of the software professionals of the organization. For smaller organizations, it is desirable to have at least one full-time SEPG professional with the part-time support of other professionals on working groups. At the very least, one person should have a substantial part-time assignment as SEPG leader and several others have smaller part-time assignments as SEPG and implementation team members. The principal caution is that part-time not be allowed to shrink to zero-time. These activities must be scheduled and time accounted to management just as for a project assignment.

- Where does the staff come from?

  Each process group member should meet this fundamental set of qualifications:

- be a regular employee of the site, with experience within the software projects that will be served by the process group.
- have application domain expertise.
- have a basic understanding of the software development process.
- have knowledge of divisional software development methods, tools, and practices.

A SEPG having members experienced in software design, testing, management, tools, maintenance, proposal preparation, and quality assurance will be able to work credibly with the steering committee, a range of working groups, and other software practitioners.

SEPG members should have worked in several different contexts and have experience with multiple frames of reference. In addition, process group members need consulting skills and strong oral and written communication skills. They must be able to communicate effectively with peers and superiors, and must have a reputation for productive work relationships with both. An open mind is essential, as group members must be able to rapidly understand and appreciate many different technical and organizational contexts.

- Who should head the SEPG?

  The key criteria for the selection of the SEPG leader are listed below. A potential candidate:

  - should be enthusiastic about leading the change process.
  - must be both technically and politically capable of understanding the problems and ensuring that effective solutions are implemented.
  - needs the respect of the people they are to deal with.
  - must have management's confidence and support or they will not act with the assurance needed to get wide cooperation and acceptance.

- What should be the length of membership?

  Tenure of two to three years is recommended for SEPG members. Membership should be staggered, allowing a month or two of overlap as members are replaced. Ideally, SEPG members should come from and return to line organizations: i.e., organizations building or revising software. If it is a small organization, SEPG members may continue as long as they are needed.

- Where should the SEPG report?

  The SEPG could, for example, report to the same executive reporting point as SQA, the computing center, SCM, or software technology. The SEPG must not report to line development management or to the SQA organization.

REASONING--In either case their role would likely become one of taking sides in the traditional SQA/development conflicts and their own focus on process improvement would suffer.

E.3. SEPG Evaluation

The SEPG could be judged on how effectively it applies an improvement framework to its own work. This should show a clear and succinct picture of what the group is doing and where they stand with respect to the following criteria:

1. Does the SEPG have a plan for its work, a tracking system, and means to retain and control its work products?
2. Have the SEPG professionals established a basic framework for their own work, including standards, procedures, and a review program?
3. Does the SEPG measure the productivity and quality of its own work? This, for example, might include metrics such as workload factors for training, consultation, process development, and administration.
4. Does the SEPG regularly assess their own activities for improvement opportunities and incorporate them in their own working process?

For the SEPG to be successful, this team must practice all the management practices they are asking the software staff to implement.

E.4. SEPG Tasks

The initial emphasis of the SEPG should be on project planning and project management. As with all improvement teams the most important single guideline is for the SEPG to limit its focus to those tasks it can handle reasonably quickly and effectively. (The low hanging fruit.)

The following is a list of the tasks that SEPGs' normally attack:

- Obtain and maintain support for process improvements by practitioners and all levels of management.
- Establish and lead improvement teams.
- Facilitate internal software process assessments of projects (these assessments are a gap analysis process utilizing a framework from the Software Engineering Institute).
- Work with line managers affected by improvements.
- Arrange for any training related to improvements.
- Track and report status on improvement efforts.

- Facilitate creation and maintenance of process definitions and a process repository/ database.
- Provide process consultation to practitioners and management.
- Train and practice so that the SEPG is continuously improving.

E.5. NWC Software Process Improvement Group

It is suggested that the DOE Quality Managers, with the Engineering Managers, form an NWC-wide Software Process Improvement Group. The tasks that this team will take on are similar to the sites' SEPGs, however, the emphasis will be on inter-site communication, the review of site progress on implementing action plans from the software process assessments, and the continuous improvement of each site's software engineering process framework. This team should consist of members from the sites. Site SEPG team leaders are expected to represent the site on the NWC Software Process Improvement Group.

F. Guidelines for Selecting Assessment Team Members and Project Selection

This section describes the staff involved with an formal assessment. Much of the information here may also be applied to a self-assessment.

F.1. Assessment Participants And Supporting Personnel

Organization codes:

Non-Local - Normally provided by another organization or an SEI licensed vendor

Local - Provided by the software organization being assessed

Assessment Participants

Role	Main Responsibility	Organization	Commitment
Assessment Team Leader	Teaches, facilitates, and leads the assessment; presents final findings; an assessment team member	Non-Local	28 days minimum (will depend on the assessment experience)
Assessment Team Coordinator	Assists Assessment Team Leader; handles logistics; an assessment team member	Local	21 days
Assessment Team Member	Performs the assessment; creates final findings	Non-Local/Local **	13 days/ Assessment Team Member
Project Leader *	Provides data for the selected projects during interviews	Local	1 day/Project Leader
Functional Area Representative	Software practitioner to participate in discussion groups	Local	1 day/Functional Area Representative

Non-participant Support Personnel

Role	Main Responsibility	Organization	Commitment
Senior Management *	Senior Executive (sponsor) and staff; speak at Opening Meeting & Final Findings Presentation	Local	1 day
Middle Management *	Review preliminary findings	Local	2 hours
Administrative Support	Provides support to produce Final Findings Presentation	Local	2 days

* Actual titles used in the organization may vary; refer to the descriptions in Section F.2.

** Assessment Team Members can also be provided from other organizations.

The actual number of Assessment Team Members and Functional Area Representatives selected depends on the size of the software organization. At least 25% of the software organization being assessed should participate - participants include the Assessment Team Coordinator, Assessment Team Members, Project Leaders, and Functional Area Representatives. ***A person may not participate in more than one role.***

F.2. Roles And Responsibilities For Assessment Participants

Assessment Team Leader

Although the Assessment Team Leader is usually provided by an SEI licensed vendor or another organization, it helps to know the roles and responsibilities of this position to better enable you to select the remaining assessment participants. Also, part of software improvement process is to have a follow-on assessment as improvements are made in the organization. A re-assessment is conducted approximately 18 - 24 months after this initial assessment and could be lead by organization personnel.

Minimum qualifications for an Assessment Team Leader include:

- Considerable software experience (>10 yrs since BS degree),
- Degree in Computer Science, Electrical Engineering, or related technical field (BS or higher),
- Experience as an assessment team member,
- Ability to manage small groups and keep focused, and
- Ability to manage expectations and present convincingly.

The Assessment Team Leader is an experienced software engineer, trained and experienced in conducting assessments, and who is responsible for the overall success of the assessment.

Responsibilities include:

- Approve selection of the on-site assessment team coordinator.
- Review resumes and qualifications of potential team members.
- Review the final selection of projects to be assessed.
- Ensure the right selection of participants to foster communication.
- Conduct the assessment training, including:
  - Team building exercise,
  - Assessment walkthrough with team members,
  - Assessment Participants Briefing,
  - Executive Briefing to Senior Management,
  - Having the Project Leaders fill out the SEI Questionnaire, and
  - Approval of response analysis scripts for project interviews.
- Determine roles and responsibilities for team members during the on-site period.
- Lead the assessment.
- Present the final findings to Senior Management.

Phases:	**Assessment Team Leader's Role:**
Planning	Lead person; work with Assessment Team Coordinator
Training	Lead person; train Assessment Team Members, conduct briefings, assign roles for interviews and discussions, approve response analysis scripts
Assessment	Lead person; conduct briefings, approval final findings, participate as Assessment Team Member
Final Findings Presentations	Lead person; present to Senior Management
Executive Session	Gives response to findings, provide private feedback

Assessment Team Coordinator

The Assessment Team Coordinator is provided by the organization being assessed and works closely with the Assessment Team Leader.

Qualifications for the Assessment Team Coordinator include:

- Meet all qualifications for an assessment team member,
- Considerable software experience (> 8 yr. since BS degree),
- Degree in Computer Science, Electrical Engineering, or related technical field (BS or higher),
- Knowledge of the software organization being assessed in order to prepare an organizational overview,
- Respect of the organization; someone whose opinions are sought out by peers, and
- A good working relationship with management, Assessment Team Members, Project Leaders, and Functional Area Representatives.

The Assessment Team Coordinator will be contacted by the Assessment Team Leader, together they will:

- Select remaining assessment participants:
  - Assessment Team Members to perform the assessment,
  - Functional Area Representatives for group discussions, and
  - Project Leaders for project interviews.
- Select projects to be assessed.
- Arrange for administrative support.
- Arrange conference calls or meetings with the Assessment Team Members.
- Schedule assessment briefings and meetings with the Senior Manager (assessment sponsor).
- Review suggestions for Senior Manager comments during Opening Meeting and Final Findings Presentation.

The Assessment Team Coordinator must:

- Arrange all logistics (e.g. facilities, food, lodging for out-of-town Assessment Team Members, administrative support, supplies).
- Provide an overview of the organization being assessed (present to the assessment team during training week).
- Coordinate schedules with management, Project Leaders, and Functional Area Representatives for their participation in the assessment.

Phases:	**Assessment Team Coordinator's Role:**
Planning	Assist Assessment Team Leader; handle logistics, review comments with Senior Manager, contact participants

Training	Assist Assessment Team Leader; handle logistics, participate as Assessment Team Member
Assessment	Assist Assessment Team Leader; handle logistics and administrative support, participate as Assessment Team Member
Final Findings Presentation	Assist Assessment Team Leader; handle questions from Senior Management and assessment participants
Executive Session	Gives response to findings
Follow-up	Lead person, should head site SEPG

Assessment Team Member

Assessment Team Members can be provided by the SEI licensed vendor, another organization, or the organization being assessed. For a first assessment, the Vendor or outside organization normally provides the Assessment Team Leader and up to two additional members. The remaining 2 - 4 team members are provided by the organization being assessed. The organization being assessed must provide at least one member in addition to the on-site Assessment Team Coordinator. Team members are critical to the success of the assessment and should be carefully selected.

Qualifications for team members from the organization being assessed include:

- Considerable software experience (> 5 years since degree),
- Minimum of two years experience with the organization being assessed,
- Being a practitioner, not a manager or staff member,
- Motivation to improve the organization,
- Actual working experience in one or more software life-cycle phases (the team, as a whole, should have experience in each major phase),
- Being a team player so as not to inhibit the free flow of information,
- Ability to reach decisions and support the assessment findings, and
- Full-time employee (i.e. no contractors or part-timers).

Assessment Team Members should be drawn from project groups, test organizations, and support groups.

Responsibilities of team members include:

- Assist the on-site coordinator with participant selection.
- Receive training from the team leader.
- Develop/review response analysis scripts (based on the results of the SEI Questionnaire).

- Interview project leaders and facilitate discussion groups during the on-site period.
- Help generate the final findings.
- Participate in follow-on activities.

Phases:	**Assessment Team Member's Role:**
Planning	Work with on-site coordinator (Assessment Team Coordinator) as necessary
Training	Team Building, Assessment Walkthrough, Develop/review Response Analysis Scripts, attend Assessment Participants Briefing
Assessment	Conduct interviews and discussion groups; generate final findings
Final Findings Presentation	Attend; may handle questions from participants
Follow-up	Assist Assessment Team Coordinator; Assessment Team Members from the organization should serve on the local SEPG, but they should not then serve on later follow-on assignment teams. (They will be too close to the process improvement issue to be objective and will prevent a true evaluation of the organization.)

Project Leader And Project Selection

The Assessment Team Leader and Assessment Team Coordinator will select the four projects to be assessed. Project selection is based on:

- Each project being in a different life-cycle phase.
- Representative of your site's business.
- Avoiding projects that are extreme in size and number of problems, use very old/new languages or design technologies, or are very early in the life-cycle. Mature projects offer more insight into the organization.

Although it may not be possible for each project to meet all of the above criteria, the projects selected should best represent the essential characteristics of the software organization. Non-typical projects tend to skew assessment results toward isolated, rather than inherent, findings.

The project leader should:

- Have the lead responsibility and authority for the selected project.

- Not be participating in the assessment or be a project leader on another selected project.

Phases:	**Project Leader's Role:**
Planning	Fill out project questionnaire
Training	Assessment Participants Briefing, fill out SEI Questionnaire
Assessment	Opening Meeting, Project Interviews (Rounds 1 and 2), and dry-run
Final Findings Presentation	Attendance Required
Follow-up	As needed, may sign-up for a team to work specific findings, work with or serve on SEPG

Functional Area Representative Groups

Functional Area Representatives are the software practitioners from the software organization being assessed. They should be representatives from each life-cycle and support group. Examples of potential Functional Area Representative groups are:

- Quality Assurance,
- Configuration Management,
- Software Release,
- Software Integration and Test,
- Design and Code, and
- Requirements and Design.

If your software organization does not support all of these groups, different Functional Area Representative groups can be formed. For example, a maintenance organization could have members from Transition and Configuration Management, Acceptance Test, Quality Assurance, and Requirements Analysis.

A Functional Area Representative should have the following qualifications:

- A technical professional, not a manager,
- A software practitioner,
- Respected by peers as an opinion leader,
- Considered to have expertise in the group being represented,
- Have at least one year experience in the software organization being assessed, and
- Someone who encourages and participates in the free flow of conversation.

Representatives should be selected from the total population of the software organization, not just the selected projects. A person should not serve in more than one Functional Area Representative group, even if their job spans multiple life-cycle phases.

Functional Area Representatives can either be selected or volunteer, as long as the selection criteria are met.

Phases:	**Functional Area Representative's Role:**
Planning	Contacted by Assessment Team Coordinator to participate
Training	Assessment Participants Briefing
Assessment	Opening Meeting, Functional Area Representative group discussion, and dry-run
Final Findings Presentation	Attendance required
Follow-up	As needed; may sign-up for a team to work specific findings, work with or serve on SEPG

Although the following do not actually participate in the assessment, their support is critical to the success of the assessment and ongoing improvement efforts.

Senior Management

This group includes the Senior Manager and management staff. The Senior Manager is the person having authority over the organization being assessed. No one who participates in the assessment can be outside the Senior Manager's span of control. **The Senior Manager is the one with the authority to allocate funds for improvement actions** -- i.e. the assessment sponsor, the one who owns the software process.

NOTE: MANDATORY means the Senior Manager MUST ATTEND (do not send a replacement). If the Senior Manager does not attend, the assessment should not continue.

Phases:	**Senior Management's Role:**
Planning	Prepare Opening Meeting and Final Findings Presentation comments during on-site period (review with Assessment Team Coordinator)
Training	MANDATORY attendance at Executive Briefing
Assessment	MANDATORY attendance at Opening Meeting; give comments
Final Findings Presentation	MANDATORY attendance; (includes all the Assessment Team), thanks the team and initial response to findings
Executive Session	MANDATORY attendance; (includes other management if

	desired), gives response to findings, private session to allow the Assessment Team Leader to provide their private feedback
Follow-up	Provide necessary resources and support

Middle Management

Middle management reviews the preliminary findings dry-run as their support and buy-in is critical for the ultimate success of the assessment.

Phases:	**Middle Management's Role:**
Planning	Provide resources for Assessment Team Coordinator
Training	Executive Briefing (if requested)
Assessment	Opening Meeting and dry-run
Final Findings Presentation	Attend; show support for Senior Manager
Follow-up	Provide necessary resources and support

Administrative Support

Turnaround for the dry-run and Final Findings Presentation needs to be done quickly; therefore an administrative support person with 24-hour-a-day access to a desktop publishing workstation (PC or equivalent) with a full-screen editor (WordPerfect, Microsoft Word, or equivalent) is required during training and the on-site period. The team leader will provide the presentation template during training week. The administrative support person should be prepared to stay as long as necessary on the third and fourth days of the on-site period.

Phases:	**Administrative Support's Role:**
Planning	Support for Assessment Team Coordinator (if needed)
Training	Copy Final Findings Presentation template to workstation; prepare copies for Executive Briefing
Assessment	Edit and prepare copies for dry-run
Final Findings Presentation	Edit and prepare copies for Final Findings Presentation
Follow-up	Support for Assessment Team Coordinator (if needed)

G. Software Engineering Process Maturity Levels, SEI Questions and Key Process Areas

G.1. Overview

Both assessment methods presented in this paper are based upon the SEI assessment process and their Capability Maturity Model (CMM). The purpose of this section is to provide the reader with background information on both the SEI assessment process, the CMM and questions.

An SEI Assessment is "an appraisal by a trained team of software professionals to determine the state of an organization's current software process, to determine the high-priority software process-related issues facing an organization, and to obtain the organizational support for software process improvement."

The SEI assessment process has two major outputs:

1. Identify the maturity of the organization being assessed and
2. Most importantly, identify the major areas which need improvement and create the beginning of an improvement strategy.

Although software engineers and managers often know their problems in great detail, they may disagree on which improvements are most important. Without an organized strategy for improvement, it is difficult to achieve consensus between management and the staff on which improvement activities to undertake first. To achieve lasting results from process improvement efforts, it is necessary to design an evolutionary path that increases an organization's software process maturity in stages. The SEI software process maturity framework orders these stages so that improvement at each stage provides the foundation on which to build improvements undertaken at the next stage. Thus, an improvement strategy drawn from the SEI software process framework provides a start for continuous process improvement. It guides advancement and identifies deficiencies in the organization; it is not intended to provide a quick fix for projects in trouble.

The SEI CMM identifies 5 levels of maturity. Those levels are identified in the chart which follows. In order for an organization to be assessed at a particular level they must have fully implemented the Key Process Areas identified. For example, to be a level 2 an organization must have satisfied the maturity questionnaire questions of the Key Process Areas: Requirements Management, Software Project Planning, Software Project Tracking and Oversight, Software Subcontract Management, Software Configuration Management, and Software Quality Assurance.

Lower levels of process maturity translate into increased risk. Level 3 is the level at which the risk associated with an immature process diminishes. The chart and words that follow provide an overview of the CMM:

SEI Process-Maturity Framework

LEVEL	CHARACTERISTICS	FOCUS	KEY PROCESS AREAS
5 Optimizing	• Improvement fed back into process • Data is automated and used to identify weakest process elements • Numerical evidence used to justify application of technology to tasks • Rigorous defect - cause analysis and defect prevention	Continuous process improvement	• Defect Prevention • Technology Change Management • Process Change Management
4 Managed	(Quantitative) • Measured process • Minimum set of quality and productivity measurements established • Process database established with resources to analyze its data and maintain it	Product and process quality	• Quantitative Process Management • Software Quality Management
3 Defined	(Qualitative) • Process defined and institutionalized • Software Engineering Process Group established to lead process improvement	Engineering process	• Organizational Process Focus • Organizational Process Definition • Training Program • Integrated Software Management • Software Product Engineering • Intergroup Coordination • Peer Reviews
2 Repeatable	(Intuitive) • Process dependent on individuals • Established basic project controls • Strength in doing similar work, but faces major risk when presented with new challenges • Lacks orderly framework for improvement	Project Management	• Requirements Management • Software Project Planning • Software Project Tracking and Oversight • Software Subcontract Management • Software Configuration Management • Software Quality Assurance
1 Initial	(Ad hoc/chaotic process) • No formal procedures, cost estimates, project plans • No management mechanism to ensure procedures are followed, tools not well integrated, and change control is lax • Senior management does not understand key issues	Heroes	

Initial: The initial environment has ill-defined procedures and controls. The organization does not consistently apply software engineering management to the process, nor does it use modern tools and technology. Level 1 organizations may have serious cost and schedule overruns.

Repeatable: At Level 2, the organization has generally learned to manage costs and schedules, the process is now repeatable. The organization uses standard methods and practices for managing software development activities such as cost estimating, scheduling, requirements changes, code changes, and status reviews.

Defined: In Level 3, the process is well characterized and reasonably well understood. The organization defines its process in terms of software engineering standards and methods, and it has made a series of organizational and methodological improvements. These specifically include design and code reviews, training programs, and increased organizational focus on software engineering. A major improvement in this phase is the establishment and staffing of software engineering process groups that focus on the software engineering process improvement and the adequacy with which it is implemented.

Managed: In Level 4, the process is not only understood, but it is quantified, measured, and reasonably well controlled. The organization typically bases its operating decisions on quantitative process data, and conducts extensive analyses of the data gathered during software engineering reviews and tests. Tools are used increasingly to control and manage the design process as well as to support data gathering and analysis. The organization is learning to predict trends in process and product quality within limits and estimate expected errors with reasonable accuracy.

Optimized: At Level 5, organizations have not only achieved a high degree of control over their process, they have a major focus on improving and optimizing its operation. This includes more sophisticated analyses of the error and cost data gathered during the process as well as the introduction of comprehensive error cause analysis and prevention studies. The data on the process is used interactively to improve the process and achieve optimum performance.

It is common for an organization to have some attributes of all of the stages.

The following section comes from a SEI Technical Report (CMU/SEI-87-TR-23). It defines each of the process maturity levels and the associated questions that were part of the assessment process at that time. The questions with a * are considered to be of greater importance for the indicated maturity level.

In order to achieve clarity in the questions, many of the terms used have been given specific explanatory definitions in the glossary (section G.5). Each use of a glossary term in the questions is italicized. Adherence to these definitions is essential for proper and consistent assessments.

G.2. Determining Process Maturity

To determine a level of process maturity, the following procedure is used. This procedure requires successive qualifications at each level.

1. Determine the percentage of affirmative answers to all Level 2 questions and to the asterisked questions for Level 2. If the percentage of affirmative answers to all questions is at least 80% and the percentage of affirmative answers to asterisked questions is at least 90%, the organization has qualified at Level 2; otherwise, it is at Level 1. If Level 2 is achieved, go to the next step.
2. Determine the percentage of affirmative answers to all Level 2 and 3 questions combined and to the asterisked questions for Levels 2 and 3 combined. Again, if the percentage of affirmative answers to all questions is at least 80% and the percentage of affirmative answers to asterisked questions is at least 90%, the organization qualifies at Level 3, otherwise, it is at Level 2. If it qualifies at Level 3, this procedure is repeated combining Level 2, 3, and 4 answers, again requiring 80% for all questions and 90% for asterisked questions. If the organization qualifies at Level 4, the assessment for Level 5 combines 2, 3, 4, and 5 answers, again using 80% and 90% as the criteria.
3. Determine the level for the organization as a whole by averaging the levels of projects assessed.

G.3. Maturity Levels

Level 1 – Initial Process

The initial environment has ill–defined procedures and controls. While positive responses to some of the organizational questions are likely, the organization does not consistently apply software engineering management to the process, nor does it use modern tools and technology.

Level 2 – Repeatable Process

At Maturity Level 2, the organization uses standard methods and practices for managing software development activities such as cost estimating, scheduling, requirements changes, code changes, and status reviews. The organization will provide positive responses to most of the following questions.

1.1.1 For each project involving software development, is there a designated software manager?

1.1.2 Does the project software manager report directly to the project (or project development) manager?

*1.1.3	Does the Software Quality Assurance (SQA) function have a management reporting channel separate from the software development project management?
*1.1.6	Is there a software configuration control function for each project that involves software development?
1.2.2	Is there a required training program for all newly appointed development managers designed to familiarize them with software project management?
1.3.1	Is a *mechanism* used for maintaining awareness of the state–of–the–art in software engineering technology?
*2.1.3	Is a *formal procedure* used in the management review of each software development prior to making contractual commitments?
2.1.4	Is a *formal procedure* used to assure periodic management review of the status of each software development project?
2.1.5	Is there a *mechanism* for assuring that software subcontractors, if any, follow a disciplined software development *process*?
2.1.7	For each project, are independent audits conducted for each step of the software development *process*?
2.1.9	Are coding *standards* applied to each software development project?
*2.1.14	Is a *formal procedure* used to make estimates of software size?
*2.1.15	Is a *formal procedure* used to produce software development schedules?
*2.1.16	Are *formal procedures* applied to estimating software development schedules?
2.1.17	Is a *mechanism* used for ensuring that the software design teams understand each software requirement?
2.2.1	Are software staffing profiles maintained of actual staffing versus planned staffing?
*2.2.2	Are profiles of software size maintained for each software configuration item, over time?
*2.2.4	Are statistics on software code and test errors gathered?
2.2.7	Are profiles maintained of actual versus planned software units designed, over time?
2.2.8	Are profiles maintained of actual versus planned software units completing unit testing, over time?

2.2.9 Are profiles maintained of actual versus planned software units integrated, over time?

2.2.10 Are target computer memory utilization estimates and actuals tracked?

2.2.11 Are target computer throughput utilization estimates and actuals tracked?

2.2.12 Is target computer I/O channel utilization tracked?

2.2.16 Are software trouble reports resulting from testing tracked to closure?

2.2.18 Is test progress tracked by deliverable software component and compared to the plan?

2.2.19 Are profiles maintained of software build/release content versus time?

*2.4.1 Does senior management have a *mechanism* for the regular review of the status of software development projects?

2.4.5 Is a *mechanism* used for regular technical interchanges with the customer?

*2.4.7 Do software development first–line managers sign off on their schedules and cost estimates?

*2.4.9 Is a *mechanism* used for controlling changes to the software requirements?

*2.4.17 Is a *mechanism* used for controlling changes to the code? (Who can make changes and under which circumstances?)

2.4.20 Is there a *mechanism* for assuring that regression testing is routinely performed?

Level 3 – Defined Process

At Maturity Level 3, the organization not only defines its process in terms of software engineering standards and methods, it also has made a series of organizational and methodological improvements. These specifically include design and code reviews, training programs for programmers and review leaders, and increased organizational focus on software engineering. A major improvement in this phase is the establishment and staffing of a software engineering process group that focuses on the software engineering process and the adequacy with which it is implemented. In addition to the questions for Level 2, organizations at Level 3 will respond “yes” to most of the following questions.

1.1.4 Is there a designated individual or team responsible for the control of software interfaces?

1.1.5 Is software system engineering represented on the system design team?

*1.1.7 Is there a software engineering *process group* function?

1.2.1 Does each software developer have a private computer–supported workstation/terminal?

*1.2.3	Is there a required software engineering training program for software developers?
1.2.4	Is there a required software engineering training program for first–line supervisors of software development?
*1.2.5	Is a formal training program required for design and code *review leaders*?
1.3.2	Is a *mechanism* used for evaluating technologies used by the organization versus those externally available?
*2.1.1	Does the software organization use a standardized and documented software development *process* on each project?
2.1.2	Does the standard software development *process* documentation describe the use of tools and techniques?
2.1.6	Are *standards* used for the content of software development files/folders?
2.1.8	Is a *mechanism* used for assessing existing designs and code for reuse in new applications?
2.1.10	Are *standards* applied to the preparation of unit test cases?
2.1.11	Are code maintainability *standards* applied?
2.1.18	Are man–machine interface *standards* applied to each appropriate software development project?
*2.2.3	Are statistics on software design errors gathered?
*2.2.15	Are the action items resulting from design reviews tracked to closure?
*2.2.17	Are the action items resulting from code reviews tracked to closure?
2.4.3	Is a *mechanism* used for identifying and resolving system engineering issues that affect software?
2.4.4	Is a *mechanism* used for independently calling integration and test issues to the attention of the project manager?
*2.4.6	Is a *mechanism* used for ensuring compliance with the software engineering *standards*?
2.4.8	Is a *mechanism* used for ensuring traceability between the software requirements and top–level design?
2.4.11	Is a *mechanism* used for ensuring traceability between the software top–level and detailed designs?
*2.4.12	Are internal software design reviews conducted?
*2.4.13	Is a *mechanism* used for controlling changes to the software design?
2.4.14	Is a *mechanism* used for ensuring traceability between the software detailed design and the code?

2.4.15 Are formal records maintained of unit (module) development progress?

*2.4.16 Are software code reviews conducted?

2.4.18 Is a *mechanism* used for configuration management of the software tools used in the development *process*?

*2.4.19 Is a *mechanism* used for verifying that the samples examined by Software Quality Assurance are truly representative of the work performed?

*2.4.21 Is there a *mechanism* for assuring the adequacy of regression testing?

2.4.22 Are formal test case reviews conducted?

Level 4 – Managed Process

At Maturity Level 4, the organization typically bases its operating decisions on quantitative process data, and conducts extensive analyses of the data gathered during software engineering reviews and tests. Tools are used increasingly to control and manage the design process as well as to support data gathering and analysis. The organization is learning to project expected errors with reasonable accuracy. In addition to questions for Levels 2 and 3, organizations at Level 4 will respond "yes" to most of the following questions.

1.3.3 Is a *mechanism* used for deciding when to insert new technology into the development *process*?

*1.3.4 Is a *mechanism* used for managing and supporting the introduction of new technologies?

2.1.12 Are internal design review *standards* applied?

*2.1.13 Are code review *standards* applied?

*2.2.5 Are design errors projected and compared to actuals?

*2.2.6 Are code and test errors projected and compared to actuals?

*2.2.13 Are design and code *review coverages* measured and recorded?

*2.2.14 Is *test coverage* measured and recorded for each phase of functional testing?

*2.3.1 Has a managed and controlled *process database* been established for process metrics data across all projects?

*2.3.2 Are the *review data* gathered during design reviews analyzed?

*2.3.3 Is the error data from code reviews and tests analyzed to determine the likely distribution and characteristics of the errors remaining in the product?

*2.3.4 Are analyses of errors conducted to determine their *process* related causes?

*2.3.8 Is *review efficiency* analyzed for each project?

2.3.9 Is software productivity analyzed for major *process* steps?

*2.4.2 Is a *mechanism* used for periodically assessing the software engineering *process* and implementing indicated improvements?

2.4.10 Is there a formal management *process* for determining if the prototyping of software functions is an appropriate part of the design *process*?

Level 5 – Optimized Process

At Maturity Level 5, organizations have not only achieved a high degree of control over their process, they have a major focus on improving and optimizing its operation. This includes more sophisticated analyses of the error and cost data gathered during the process as well as the introduction of comprehensive error cause analysis and prevention studies.

*1.3.5 Is a *mechanism* used for identifying and replacing obsolete technologies?

*2.3.5 Is a *mechanism* used for error cause analysis?

*2.3.6 Are the error causes reviewed to determine the *process* changes required to prevent them?

*2.3.7 Is a *mechanism* used for initiating error prevention actions?

G.4. Key Process Areas

Except for Level 1, each maturity level is decomposed into several Key Process Areas that indicate the areas an organization should focus on to improve its software process. Key Process Areas identify the issues that must be addressed to achieve a maturity level.

Each *Key Process Area* identifies a cluster of related activities that, when performed collectively, achieve a set of goals considered important for enhancing process capability. The Key Process Areas have been defined to reside at a single maturity level. The path to achieving the goals of a Key Process Area may differ across projects based on differences in application domains or environments. Nevertheless, all the goals of a Key Process Area must be achieved for the organization to satisfy that Key Process Area. When the goals of a Key Process Area are accomplished on a continuing basis across projects, the organization can be said to have institutionalized the process capability characterized by the Key Process Area.

The adjective “key” implies that there are process areas (and processes) that are not key to achieving a maturity level. The CMM does not describe all the process areas in detail that are involved with developing and maintaining software. Certain process areas have

been identified as key determiners of process capability; these are the ones described in the CMM.

Although other issues affect process performance, the Key Process Areas were identified because of their effectiveness in improving an organization's software process capability. They may be considered the requirements for achieving a maturity level. The Table in G.1 displays the Key Process Areas for each maturity level. To achieve a maturity level, the Key Process Areas for that level must be satisfied. To satisfy a Key Process Area, each of the goals for the Key Process Area must be satisfied. The *goals* summarize the key practices of a Key Process Area and can be used to determine whether an organization or project has effectively implemented the Key Process Area. The goals signify the scope, boundaries, and intent of each process area.

The specific practices to be executed in each Key Process Area will evolve as the organization achieves higher levels of process maturity. For instance, many of the project estimating capabilities described in the Software Project Planning Key Process Area at Level 2 must evolve to handle the additional project data available at Levels 3, 4, and 5. Integrated Software Management at Level 3 is the evolution of Software Project Planning and Software Project Tracking and Oversight at Level 2 as the project is managed using a defined software process.

The Key Process Areas of the CMM represent one way of describing how organizations mature. These Key Process Areas were defined based on many years of experience in software engineering and management and over five years of experience with software process assessments and software capability evaluations.

The Key Process Areas at Level 2 focus on the software project's concerns related to establishing basic project management controls. Descriptions of each of the Key Process Areas for Level 2 are given below:

- The purpose of Requirements Management is to establish a common understanding between the customer and the software project of the customer's requirements that will be addressed by the software project. This agreement with the customer is the basis for planning (as described in Software Project Planning) and managing (as described in Software Project Tracking and Oversight) the software project. Control of the relationship with the customer depends on following an effective change control process (as described in Software Configuration Management).
- The purpose of Software Project Planning is to establish reasonable plans for performing the software engineering and for managing the software project. These plans are the necessary foundation for managing the software project (as described in Software Project Tracking and Oversight). Without realistic plans, effective project management cannot be implemented.

- The purpose of Software Project Tracking and Oversight is to establish adequate visibility into actual progress so that management can take effective actions when the software project's performance deviates significantly from the software plans.
- The purpose of Software Subcontract Management is to select qualified software subcontractors and manage them effectively. It combines the concerns of Requirements Management, Software Project Planning, and Software Project Tracking and Oversight for basic management control, along with necessary coordination of Software Quality Assurance and Software Configuration Management, and applies this control to the subcontractor as appropriate.
- The purpose of Software Quality Assurance is to provide management with appropriate visibility into the process being used by the software project and of the products being built. Software Quality Assurance is an integral part of most software engineering and management processes.
- The purpose of Software Configuration Management is to establish and maintain the integrity of the products of the software project throughout the project's software life cycle. Software Configuration Management is an integral part of most software engineering and management processes.

The Key Process Areas at Level 3 address both project and organizational issues, as the organization establishes an infrastructure that institutionalizes effective software engineering and management processes across all projects. Descriptions of each of the Key Process Areas for Level 3 are given below:

- The purpose of Organization Process Focus is to establish the organizational responsibility for software process activities that improve the organization's overall software process capability. The primary result of the Organization Process Focus activities is a set of software process assets, which are described in Organization Process Definition. These assets are used by the software projects, as is described in Integrated Software Management.
- The purpose of Organization Process Definition is to develop and maintain a usable set of software process assets that improve process performance across the projects and provide a basis for cumulative, long–term benefits to the organization. These assets provide a stable foundation that can be institutionalized via mechanisms such as training, which is described in Training Program.
- The purpose of Training Program is to develop the skills and knowledge of individuals so they can perform their roles effectively and efficiently. Training is an organizational responsibility, but the software projects should identify their needed skills and provide the necessary training when the project's needs are unique.

- The purpose of Integrated Software Management is to integrate the software engineering and management activities into a coherent, defined software process that is tailored from the organization's standard software process and related process assets, which are described in Organization Process Definition. This tailoring is based on the business environment and technical needs of the project, as described in Software Product Engineering. Integrated Software Management evolves from Software Project Planning and Software Project Tracking and Oversight at Level 2.
- The purpose of Software Product Engineering is to consistently perform a well–defined engineering process that integrates all the software engineering activities to produce correct, consistent software products effectively and efficiently. Software Product Engineering describes the technical activities of the project, e.g., requirements analysis, design, code, and test.
- The purpose of Intergroup Coordination is to establish a means for the software engineering group to participate actively with the other engineering groups so the project is better able to satisfy the customer's needs effectively and efficiently. Intergroup Coordination is the interdisciplinary aspect of Integrated Software Management that extends beyond software engineering; not only should the software process be integrated, but the software engineering group's interactions with other groups must be coordinated and controlled.
- The purpose of Peer Reviews is to remove defects from the software work products early and efficiently. An important corollary effect is to develop a better understanding of the software work products and of the defects that can be prevented. The peer review is an important and effective engineering method that is called out in Software Product Engineering and that can be implemented via Fagan–style inspections, structured walkthroughs, or a number of other collegial review methods.

The Key Process Areas at Level 4 focus on establishing a quantitative understanding of both the software process and the software work products being built. The two Key Process Areas at this level, Quantitative Process Management and Software Quality Management, are highly interdependent, as is described below:

- The purpose of Quantitative Process Management is to control the process performance of the software project quantitatively. Software process performance represents the actual results achieved from following a software process. The focus is on identifying special causes of variation within a measurably stable process and correcting, as appropriate, the circumstances that drove the transient variation to occur. Quantitative Process Management adds a

comprehensive measurement program to the software work products described in Software Product Engineering.

The Key Process Areas at Level 5 cover the issues that both the organization and the projects must address to implement continuous and measurable software process improvement. Descriptions of each of the Key Process Areas for Level 5 are given below:

- The purpose of Defect Prevention is to identify the causes of defects ad prevent them from recurring. The software project analyzes defects, identifies their causes, and changes its defined software process, as is described in Integrated Software Management. Process changes of general value are transitioned to other software projects, as is described in Process Change Management.
- The purpose of Technology Change Management is to identify beneficial new technologies (i.e., tools, methods, and processes) and transfer them into the organization in an orderly manner, as is described in Process Change Management. The focus of Technology Change Management is on performing innovation efficiently in an ever–changing world.
- The purpose of Process Change Management is to continually improve the software processes used in the organization with the intent of improving software quality, increasing productivity, and decreasing the cycle time for product development. Process Change Management takes the incremental improvements of Defect Prevention and the innovative improvements of Technology Change Management and makes them available to the entire organization.

G.5. Glossary

This glossary should be used in conjunction with the IEEE Standard Glossary of Software Engineering Terminology (ANSI/IEEE STD 610.12-1990) published by the Institute of Electrical and Electronic Engineers, February, 1991. Wherever possible, common software engineering terminology has been used. Where terms in this document are not included in the IEEE Standard Glossary or have special meaning in the context used here, they are described in this glossary.

contractor evaluation - A process by which a contracting organization uses the results of contractor assessments and other information to determine the relative capability of contractors.

error prevention analysis - A process that is typically conducted by a working group of software engineering professionals who developed the code in question. It is an objective assessment of each error, its potential cause, and the steps to be taken to prevent it. While placing blame is to be avoided, such questions as mistakes, adequacy of education

and training, proper tools capability, and support effectiveness are appropriate areas for analysis.

formal procedure - A documented series of steps with guidelines for use.

mechanism - A means or technique whereby the performance of a task, procedure, or process is assured. The mechanism may involve several organizational elements, and its documentation may include some combination of function statements, operating plans, position descriptions, and/or formal procedures. The documentation defines what should be performed, how it should be performed, and who is accountable for the results.

process - A systematic series of mechanisms, tasks, and/or procedures directed towards an end. The software engineering process documentation defines the sequence of steps used to produce a finished product. Each step is described as a task that is performed by using a software engineering methodology or an administrative procedure, and it prescribes the automated tools and techniques to be used.

process data - The data that is gathered about the software engineering process. It typically includes review, test, and resource data by process phase and change activity. To be most meaningful, this data should be associated with the process documentation, the tools and methods used, and the characteristics of the product being produced.

process database - A repository into which all process data is entered. It is a centralized resource managed by the process group. Centralized control of this database ensures that the process data from all projects are permanently retained and protected.

process group - The software engineering process group is composed of specialists concerned with the process used by the development organization for software development. Its typical functions include defining and documenting the process, establishing and defining metrics, gathering data, assisting projects in analyzing data, and advising management on areas requiring further attention. The process group typically conducts quarterly management reviews on process status and may provide review leaders.

process metrics - Those measurements established for each step in the software engineering process that are used to determine its effectiveness. The metrics define the results of each process stage and relate them to the resources expended, errors introduced, errors removed, and various coverage, efficiency, and productivity indicators.

review coverage - The degree to which all code in a software product has been reviewed. It is typically stated as a percentage and measures the percentage of the lines of executable code or design statements evaluated by the review process.

review data - The data is gathered from design or code reviews. This data is of two types. The first, concerning the review process, typically includes preparation time, lines of code per hour of preparation time, errors identified during preparation (by category),

hours per error found in preparation, review time, lines of code (or design statements) reviewed, code (or design statements) reviewed per hour, and errors found per review man–hour (by category). The second type, product data from the review, typically includes errors found per line of code (or design statement), action items identified from each review, action items closed for each review, items needing re–review, and re–reviews conducted.

review efficiency - The percentage of errors found through the review process. It is typically stated as a percentage and is calculated by dividing the total errors found during review by the total errors found by both review and test through the completion of product and system integration test. It does not include those errors found during acceptance test of field usage.

review leader - Typically a member of the process or assurance group who is thoroughly trained in the review process. The review leader's role is to ensure that the participants are properly prepared and that the review is efficiently and thoroughly conducted. The review leader is responsible for recording review data, making sure that the actions resulting from the review are completed, and for conducting re–reviews where appropriate.

standard - An approved, documented, and available set of criteria used to determine the adequacy of an action or object.

test coverage - The amount of code actually executed during the test process. It is stated as a percentage of the total instructions executed or paths traversed.

This page is left intentionally blank.

H. References

Bollinger, T.B., and McGowan, C., *A Critical Look at Software Capability Evaluations,* IEEE Software, July, 1991, pp. 25-41.

Dion, IEEE Software, July 1992 and July 1993.

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

Humphrey, W.S., and Synder, T.R., *Software Process Improvement at Hughes Aircraft,* IEEE Software, July, 1991, pp. 11-23.

Humphrey, W.S., and Curtis, B., *Comments on 'A Critical Look',* IEEE Software, July, 1991, pp. 42-46.

Humphrey, W.S., *Characterizing the Software Process: A Maturity Framework,* IEEE Software, March, 1988, pp. 73-79.

Humphrey, W.S., and Sweet, W.L., *A Method for Assessing the Software Engineering Capability of Contractors,* Technical Report, CMU/SEI-87-TR-23.

Humphrey, W.S., and Kitson, D.H., *The State of Software Engineering Practice: A Preliminary Report,* Technical Report, CMU/SEI-89-TR-1.

Humphrey, W.S., and Kitson, D.H., *The Role of Assessment in Software Process Improvement,* Technical Report, CMU/SEI-89-TR-3

Olson, T.G., Humphrey, W.S., and Kitson, D.H., *Conducting SEI-Assisted Software Process Assessments,* Technical Report, CMU/SEI-89-TR-7.

Paulk, M.C. and Curtis, B., *Capability Maturity Model for Software*, Technical Report, CMU/SEI-91-TR-24, August, 1991.

Paulk, M.C. et. al., *Capability Maturity Model for Software Version 1.1*, Technical Report, CMU/SEI-93-TR-24, ESC-TR-93-177, February, 1993.

Paulk, M.C., Weber, C.V., Garcia, S.M., Chrissis, M.B., and Bush, M., *Key Practices of the Capability Maturity Model,* Technical Report, CMU/S-93-TR-025.

SEI CMU, *Software Process Maturity Questionnaire,* Working Draft - MQ V.66a, August, 1991.

Requirement Documents

QC-1 - *DOE/AL Quality Criteria (QC-1).*

QC-2 - *DOE/AL Quality Criteria (QC-2).*

5700.6C Criteria - *DOE 5700.6C, Quality Assurance* section 9.b.

1330.1D - *DOE 1330.1D, Computer Software Management*

NQA-2 Part 2.7 - ASME NQA-2-1989 Edition *Quality Assurance Requirements for Nuclear Facility Applications*. Part 2.7 *Quality Assurance Requirements of Computer Software for Nuclear Facility Applications* contained in ASME NQA-2a-1990 Addenda.

ISO9001 - International Standard - Quality systems - Model for quality assurance in design, development, production, installation and servicing.