

**Special Report  
CMU/SEI-94-SR-7**

# **Maturity Questionnaire**

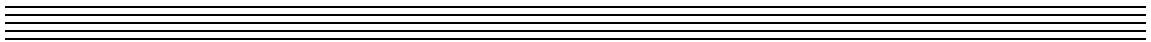
**David Zubrow  
William Hayes  
Jane Siegel  
Dennis Goldenson**

**June 1994**



**Special Report**  
CMU/SEI-94-SR-7  
June 1994

## Maturity Questionnaire



David Zubrow  
William Hayes  
Jane Siegel  
Dennis Goldenson  
Empirical Methods

Unlimited distribution subject to the copyright.

**Software Engineering Institute**  
Carnegie Mellon University  
Pittsburgh, Pennsylvania 15213

This report was prepared for the  
SEI Joint Program Office  
HQ ESC/AXS  
5 Eglin Street  
Hanscom AFB, MA 01731-2116

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

FOR THE COMMANDER

(signature on file)

Thomas R. Miller, Lt Col, USAF  
SEI Joint Program Office

This work is sponsored by the U.S. Department of Defense.

Copyright ©1994 by Carnegie Mellon University.

Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

Requests for permission to reproduce this document or to prepare derivative works of this document for external and commercial use should be addressed to the SEI Licensing Agent.

#### NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

This work was created in the performance of Federal Government Contract Number F19628-95-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 52.227-7013.

This document is available through Research Access, Inc., 800 Vinial Street, Pittsburgh, PA 15212. Phone: 1-800-685-6510. FAX: (412) 321-2994. RAI also maintains a World Wide Web home page. The URL is <http://www.rai.com>

Copies of this document are available through the National Technical Information Service (NTIS). For information on ordering, please contact NTIS directly: National Technical Information Service, U.S. Department of Commerce, Springfield, VA 22161. Phone: (703) 487-4600.

This document is also available through the Defense Technical Information Center (DTIC). DTIC provides access to and transfer of scientific and technical information for DoD personnel, DoD contractors and potential contractors, and other U.S. Government agency personnel and their contractors. To obtain a copy, please contact DTIC directly: Defense Technical Information Center, Attn: FDRA, Cameron Station, Alexandria, VA 22304-6145. Phone: (703) 274-7633.

Use of any trademarks in this report is not intended in any way to infringe on the rights of the trademark holder.

# Table of Contents

<b>Acknowledgments</b>	<b>iii</b>
<b>To the Reader</b>	<b>v</b>
<b>Software Process Maturity Questionnaire</b>	<b>1</b>
<b>Instruction Placard - Software Process Maturity Questionnaire</b>	<b>43</b>
<b>Change Request - Software Process Maturity Questionnaire</b>	<b>45</b>



## Acknowledgments

The maturity questionnaire could not have been developed without the generous assistance of the individuals and organizations listed below. At the test sites, we appreciate the contributions of the many participants who completed the various draft versions of the maturity questionnaire and provided us with their comments. We also acknowledge the administrative help from Elise Nawrocki, Carol McGinnis, Lori Imbesi-Schoell, and David White and the editorial assistance of Suzanne Couturiaux.

### **CMM Advisory Board**

Constance Ahara, GTE Government Systems  
Kelley Butler, US Air Force, Tinker AFB  
Harry Carl, Honeywell Corporation  
Conrad S. Czaplicki, Naval Air Development Ctr.  
Raymond Dion, Raytheon  
Suzanne Garcia, SEI  
Judah Mogilensky, Process Enhancement Partners  
Martin Owens, MITRE Corporation  
Mark Paulk, SEI  
Jenny Pixton, UNISYS  
Ron Radice, SEI  
Sue Stetak, Hewlett-Packard  
John Vu, Boeing Computer Services  
Charles Weber, Loral (IBM) FSC  
Ron Willis, Hughes Aircraft  
Brenda Zettervall, NSWC/PHD - Dam Neck

### **SCE Advisory Board**

Paul Byrnes, SEI  
Joseph Farinello, Electronic Systems Center  
Elizabeth Gramoy, NCCOSC NRaD  
Jeffrey Herman, US Army CECOM  
Fred C. Joh, Westinghouse Electric Corporation  
Jane A. Moon, Hughes Aircraft Company  
Daniel R. Orr, Defense Logistics Agency  
Marie Silverthorn, Texas Instruments  
Joan Weszka, Loral (IBM) FSC

### **SEI**

Mary Beth Chrissis  
Bill Curtis  
Patrick Hanavan\*  
Michael Konrad  
Gerald McCarty\*  
Mark Paulk

### **Test Sites and Points of Contact**

Arthur D. Little, Inc., Joseph Puffer  
Bull HN Information Systems Inc.,  
Jerilyn Marks  
Citicorp Services, Inc., Jennifer Simmons  
Computer Sciences Corporation, David Card  
Computing Devices International, Bill Prelogo  
Eastman Kodak Co., Sharon Rosansky  
EDS, Curt Curtis  
Fisher Rosemont, Gere Stetun  
Hewlett-Packard, Sue Stetak  
Hughes Missiles System, Dick Waina  
Institute for Software Process Improvement,  
Jeff Perdue  
Lockheed, Suzanne Garcia  
Lockheed Sanders, Barbara Bankeroff  
Logicon, Inc., William C. Nielson  
Loral (IBM) FSC, Mary Busby  
Loral (IBM) FSC, Charles Weber  
Master Systems, Inc., Stan Rifkin  
NSA, Glenn Plonk  
Pacific Bell, Roselyn Whitney  
pragma, Rebecca Bowerman  
Process Enhancement Partners,  
Judah Mogilensky  
Process Inc., Louise Hawthorne  
Schlumberger, Harvey Wohlwend  
Sematech, Ted Ziehe  
Software Productivity Consortium,  
Jack Hofmann  
Texas Instruments, Marie Silverthorn  
UNISYS, Linda Lindquist  
US Air Force, Tinker AFB, Kelley Butler  
US Navy Fleet Combat Direction Systems  
NSWC/PHD - Dam Neck, Brenda Zettervall

\*Special thanks to Gerald 'Mac' McCarty and E. Patrick Hanavan for their contributions to the internal review process.





## To the Reader

### Overview

This package contains a copy of the software process maturity questionnaire. It is intended for those interested in performing and learning about software process appraisals. This version differs in several important ways from its predecessor, *A Method for Assessing the Software Capability of Contractors* (CMU/SEI-87-TR-23). The most important difference is that this questionnaire is not an appraisal method itself; rather, it is one component that is used in different appraisal methods.

### Product Description

The software process maturity questionnaire (MQ) replaces the 1987 version of the maturity questionnaire, CMU/SEI-87-TR-23, in the 1994 set of SEI appraisal products. This version of the questionnaire is based on the capability maturity model (CMM) v1.1. It has been designed for use in the new CMM-based software process appraisal methods: the CMM-based appraisal for internal process improvement (CBA IPI) which is the update of the original software process assessment (SPA) method, CMM-based software capability evaluations (SCEs), and the interim profile method. This questionnaire focuses solely on process issues, specifically those derived from the CMM. The questionnaire is organized by CMM key process areas (KPAs) and covers all 18 KPAs of the CMM. It addresses each KPA goal in the CMM but not all of the key practices. By keeping the questions to only 6 to 8 per KPA, the questionnaire can usually be completed in one hour.

The MQ incorporates many changes based on customer feedback, internal and external reviews, and extensive field testing. The questionnaire includes a glossary of terms and KPA descriptions to assist respondents who may be unfamiliar with CMM terminology. Ample space is provided beneath each question to allow respondents to provide additional information regarding their answers. This space can be used by respondents to provide further explanation of their answers or references to supporting documentation.

### Intended Use of the Software Process Maturity Questionnaire

In a CBA IPI or CMM SCE, the questionnaire serves primarily to identify issues to be explored further during the on-site period. In an interim profile, the questionnaire is the primary source of information for rating process maturity. The descriptions for use of the MQ within each appraisal method are completely addressed in the documentation for each appraisal method. We strongly recommend that organizations wishing to use this maturity questionnaire contact the SEI for information about the new CMM-based appraisal methods. These new methods have been revised not only to incorporate the CMM, but also to include more specific guidance about how to rate software processes against the CMM in a reliable and repeatable manner. Information on how to contact the SEI regarding these products is provided in "Other Related SEI Products."

## **Comparison of the MQ to CMU/SEI-87-TR-23**

The MQ is not a method for appraisal as was CMU/SEI-87-TR-23 and does not come with a scoring method as did CMU/SEI-87-TR-23. Rating procedures for CBA IPI and CMM SCE are defined by the common rating framework (CRF), which is a framework for developing and defining CMM-based appraisal methods. Among the benefits that the CRF offers are increased reliability and consistency across CMM-based appraisal methods and a clear description of the risks associated with a particular appraisal method. Rating rules for the interim profile method are provided in the documentation for that method. Furthermore, because the questionnaire is not the standard against which ratings are performed, the answers to the questions need not be validated as they were in the original SPA and SCE methods which used CMU/SEI-87-TR-23. Finally, because MQ responses are only one of the sources of information considered in the CBA IPI and CMM SCE methods, MQ responses alone will not necessarily predict the outcome of these methods.

## **Reporting Appraisal Results to the SEI**

The SEI encourages organizations performing process appraisals to report their results to the SEI. Only through the willingness of the software community to report data and results to the SEI can the SEI provide community process maturity profiles, reports on the state of the practice, and other analytical services. We hope that as a user of this product you will take this request seriously and contact us. Nondisclosure agreements can be signed to provide additional assurance that your data will be kept confidential. Contact the Empirical Methods Project at 412-268-5243 for details about reporting results to the SEI.

Also, we would like your comments on the questionnaire. A change request form for submitting your comments on the questionnaire is included in this package.

## **Contents of the Package**

This package contains a copy of the software process maturity questionnaire, a placard providing instructions on the response options for the questions and a glossary of organizational terms used in the questionnaire, and a change request form.

Instructions for administering the questionnaire are included in the kits for conducting the various appraisal methods. They are not included in this package.

## **Other Related SEI Products**

The following are related SEI products. You can obtain information on these products through the Customer Relations Office at 412-268-5800 or e-mail [customer-relations@sei.cmu.edu](mailto:customer-relations@sei.cmu.edu).

Common rating framework (CRF) – The CRF will be available in the fourth quarter of 1994. It will be available from Research Access, Inc. (RAI).

Capability maturity model (CMM) – The CMM is available from RAI as CMU/SEI-93-TR-24 (Capability Maturity Model for Software, Version 1.1) and CMU/SEI-93-TR-25 (Key Practices of the CMM, Version 1.1).

CMM-based appraisal for internal process improvement (CBA IPI) – Information on this appraisal method is available from the CBA Project as a draft method description document. Kits for performing this type of appraisal will be available from Research Access, Inc. and the SEI.

CMM-based SCE – Information on this appraisal method is available from RAI and the SEI as CMU/SEI-93-TR-17 [Method Description Document (MDD) Version 1.0]. (MDD 2.0 is in draft form, to be released in the third quarter of 1994.)

Interim profile (IP) – Information on this appraisal method is available from SEI Customer Relations. A technical report on the method is available from RAI as CMU/SEI 94-TR-04. Kits for performing an IP will be made available through the training classes for the method.

If you have general questions about the SEI Process Program or would like information on our products and services, please contact the SEI Customer Relations Office at 412-268-5800.





# Software Process Maturity Questionnaire

Capability Maturity Model, version 1.1

April 1994

This document contains questions about the implementation of important software practices in your software organization. The questions are organized in groups of key process areas such as software project planning and software configuration management. A short paragraph describing each key process area precedes each group of questions. Unless directed otherwise by the person who administers this questionnaire, please answer the questions based on your knowledge and experience in your current project.

To help us better interpret your answers to the questions about software process in your organization, this document begins with questions about your own background in software work.

Please read and answer all of the questions. If you wish to comment on any questions or qualify your answers, please use the comment spaces provided.

Your answers will be held in strict confidence by the appraisal team. Specific answers will not be identified within your organization, nor in any other manner. Your name will be used for administrative purposes only: to guide the appraisal team during response analysis and help them contact you for any needed clarifications.

Thank you for your help.

Software Engineering Institute  
Carnegie Mellon University  
Pittsburgh, Pennsylvania

© Copyright 1994 Carnegie Mellon University  
This work is sponsored by the U.S. Department of Defense.

## Filling in Your Answers

We will be using optical scanning technology to enter your answers, so please print or write neatly throughout the questionnaire.

- Feel free to use the margins if you need more space for your written answers or other comments, but please don't write over the check boxes or crosshair (+) symbols.
- Please keep your marks within the check boxes. Any mark will do:
- You should use a pen with dark blue or black ink.

## Definitions of Terms

The Capability Maturity Model on which this Maturity Questionnaire is based uses a number of terms which may be used differently in your organization.

- **Organizational terms** are defined on the blue placard. You may wish to review it now, and refer to it as necessary as you complete the questionnaire.
- **Technical terms** are defined on the pages where they are used.



## Respondent Identification

*(Please specify)*

YOUR  
NAME:

TODAY'S  
DATE:

PROJECT  
NAME:

WORK  
TELEPHONE:

## Section I Respondent Background

1 Which best describes your current position? *(Please mark as many boxes as apply)*

- |  |   |
|--|---|
| <input type="checkbox"/> PROJECT OR TEAM LEADER        | <input type="checkbox"/> MANAGER  |
| <input type="checkbox"/> TECHNICAL MEMBER              | <input type="checkbox"/> SOFTWARE ENGINEERING PROCESS GROUP (SEPG) MEMBER |
| <input type="checkbox"/> OTHER <i>(Please specify)</i> |   |

2 On what activities do you currently work? *(Please mark as many boxes as apply)*

- |  |  |
|--|--|
| <input type="checkbox"/> SOFTWARE REQUIREMENTS | <input type="checkbox"/> SOFTWARE QUALITY ASSURANCE    |
| <input type="checkbox"/> SOFTWARE DESIGN       | <input type="checkbox"/> CONFIGURATION MANAGEMENT      |
| <input type="checkbox"/> CODE AND UNIT TEST    | <input type="checkbox"/> SOFTWARE PROCESS IMPROVEMENT  |
| <input type="checkbox"/> TEST AND INTEGRATION  | <input type="checkbox"/> OTHER <i>(Please specify)</i> |

3 Have you received any CMM-related training?  NO  YES *(Please describe)*

4 What is your software experience in: *(Please specify for each category)*

Your present organization? .....  YEARS

Your overall software experience? .....  YEARS

5 Have you participated in previous forms of Software Process Assessments, Software Capability Evaluations, and/or other forms of software process appraisals? *(Please mark one box)*

- NO
- YES → How many? *(Please specify for each category)*

# OF SPAs (Software Process Assessments)

# OF SCEs (Software Capability Evaluations)

# OF OTHER SEI-BASED METHODS *(Please describe briefly: e.g., mini-assessments or instant profiles)*

BASED ON NON-SEI PROCESS IMPROVEMENT WORK *(Please describe briefly: e.g., ISO 9000/9001 audit)*

## Section II Software Practices

### Instructions

1 To the right of each question there are boxes for the four possible responses: **Yes**, **No**, **Does Not Apply**, and **Don't Know**.

Check **Yes** when:

- The practice is well established and consistently performed.
  - The practice should be performed nearly always in order to be considered well-established and consistently performed as a standard operating procedure.

Check **No** when:

- The practice is not well established or is inconsistently performed.
  - The practice may be performed sometimes, or even frequently, but it is omitted under difficult circumstances.

Check **Does Not Apply** when:

- You have the required knowledge about the project or organization and the question asked, but you feel the question does not apply to the project.
  - For example, the entire section on “Software Subcontract Management” may not apply to the project if you don't work with any subcontractors.

Check **Don't Know** when:

- You are uncertain about how to answer the question.

2 Use the **Comments** spaces for any elaborations or qualifications about your answers to the questions.

3 Check one of the boxes for each question. Please answer all of the questions.



• **This Page Intentionally Left Blank** •

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. Requirements Management involves establishing and maintaining an agreement with the customer on the requirements for the software project. The agreement covers both the technical and nontechnical (e.g., delivery dates) requirements. The agreement forms the basis for estimating, planning, performing, and tracking the software project's activities throughout the software life cycle. Whenever the system requirements allocated to software are changed, the affected software plans, work products, and activities are adjusted to remain consistent with the updated requirements.

**allocated requirements** (system requirements allocated to software) - The subset of the system requirements that are to be implemented in the software components of the system. The allocated requirements are a primary input to the software development plan. Software requirements analysis elaborates and refines the allocated requirements and results in software requirements that are documented.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**software plans** - The collection of plans, both formal and informal, used to express how software development and/or maintenance activities will be performed. Examples of plans that could be included: software development plan, software quality assurance plan, software configuration management plan, software test plan, risk management plan, and process improvement plan.

**software quality assurance (SQA)** - (1) A planned and systematic pattern of all actions necessary to provide adequate confidence that a software work product conforms to established technical requirements. (2) A set of activities designed to evaluate the process by which software work products are developed and/or maintained.

**software work product** - Any artifact created as part of defining, maintaining, or using a software process, including process descriptions, plans, procedures, computer programs, and associated documentation, which may or may not be intended for delivery to a customer or end user.

		Yes	No	Does Not Apply	Don't Know
1	Are system requirements allocated to software used to establish a baseline for software engineering and management use?.....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:					
2	As the systems requirements allocated to software change, are the necessary adjustments to software plans, work products, and activities made? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:					

- |   | Yes                      | No                       | Does Not Apply           | Don't Know               |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 3 Does the project follow a written organizational policy for managing the system requirements allocated to software? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 4 Are the people in the project who are charged with managing the allocated requirements trained in the procedures for managing allocated requirements? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 5 Are measurements used to determine the status of the activities performed for managing the allocated requirements (e.g., total number of requirements changes that are proposed, open, approved, and incorporated into the baseline)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are the activities for managing allocated requirements on the project subjected to SQA review? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

The purpose of **Software Project Planning** is to establish reasonable plans for performing the software engineering activities and for managing the software project. Software Project Planning involves developing estimates for the work to be performed, establishing the necessary commitments, and defining the plan to perform the work.

**commitment** - A pact that is freely assumed, visible, and expected to be kept by all parties.

**event-driven review/activity** - A review or activity that is performed based on the occurrence of an event within the project (e.g., a formal review or the completion of a life-cycle stage).

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**software plans** - The collection of plans, both formal and informal, used to express how software development and/or maintenance activities will be performed. Examples of plans that could be included: software development plan, software quality assurance plan, software configuration management plan, software test plan, risk management plan, and process improvement plan.

	Yes	No	Does Not Apply	Don't Know
1 Are estimates (e.g., size, cost, and schedule) documented for use in planning and tracking the software project? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Do the software plans document the activities to be performed and the commitments made for the software project? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
3 Do all affected groups and individuals agree to their commitments related to the software project? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
4 Does the project follow a written organizational policy for planning a software project? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				

- |   | Yes                      | No                       | Does<br>Not<br>Apply     | Don't<br>Know            |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 5 Are adequate resources provided for planning the software project (e.g., funding and experienced individuals)?.....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are measurements used to determine the status of the activities for planning the software project (e.g., completion of milestones for the project planning activities as compared to the plan)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Does the project manager review the activities for planning the software project on both a periodic and event-driven basis? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

The purpose of **Software Project Tracking and Oversight** is to provide adequate visibility into actual progress so that management can take corrective actions when the software project's performance deviates significantly from the software plans. Corrective actions may include revising the software development plan to reflect the actual accomplishments and replanning the remaining work or taking actions to improve the performance. Software Project Tracking and Oversight involves tracking and reviewing the software accomplishments and results against documented estimates, commitments, and plans, and adjusting these plans based on the actual accomplishments and results.

**commitment** - A pact that is freely assumed, visible, and expected to be kept by all parties.

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**software plans** - The collection of plans, both formal and informal, used to express how software development and/or maintenance activities will be performed. Examples of plans that could be included: software development plan, software quality assurance plan, software configuration management plan, software test plan, risk management plan, and process improvement plan.

**software work product** - Any artifact created as part of defining, maintaining, or using a software process, including process descriptions, plans, procedures, computer programs, and associated documentation, which may or may not be intended for delivery to a customer or end user.

	Yes	No	Does Not Apply	Don't Know
1 Are the project's actual results (e.g., schedule, size, and cost) compared with estimates in the software plans? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Is corrective action taken when actual results deviate significantly from the project's software plans? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
3 Are changes in the software commitments agreed to by all affected groups and individuals? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				

- |   | Yes                      | No                       | Does Not Apply           | Don't Know               |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 4 Does the project follow a written organizational policy for both tracking and controlling its software development activities? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 5 Is someone on the project assigned specific responsibilities for tracking software work products and activities (e.g., effort, schedule, and budget)? .....                                 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are measurements used to determine the status of the activities for software tracking and oversight (e.g., total effort expended in performing tracking and oversight activities)? .....    | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Are the activities for software project tracking and oversight reviewed with senior management on a periodic basis (e.g., project performance, open issues, risks, and action items)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

The purpose of **Software Subcontract Management** is to select qualified software subcontractors and manage them effectively. Software Subcontract Management involves selecting a software subcontractor, establishing commitments with the subcontractor, and tracking and reviewing the subcontractor's performance and results. These practices cover the management of a software (only) subcontract, as well as the management of the software component of a subcontract that includes software, hardware, and possibly other system components.

**documented procedure** - A written description of a course of action to be taken to perform a given task.  
[IEEE-STD-610 Glossary]

**event-driven review/activity** - A review or activity that is performed based on the occurrence of an event within the project (e.g., a formal review or the completion of a life-cycle stage).

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

		Yes	No	Does Not Apply	Don't Know
1	Is a documented procedure used for selecting subcontractors based on their ability to perform the work? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				
2	Are changes to subcontracts made with the agreement of both the prime contractor and the subcontractor? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				
3	Are periodic technical interchanges held with subcontractors? ....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				
4	Are the results and performance of the software subcontractor tracked against their commitments?.....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				



- |   | Yes                      | No                       | Does Not Apply           | Don't Know               |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 5 Does the project follow a written organizational policy for managing software subcontracts? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are the people responsible for managing software subcontracts trained in managing software subcontracts?.....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Are measurements used to determine the status of the activities for managing software subcontracts (e.g., schedule status with respect to planned delivery dates and effort expended for managing the subcontract)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 8 Are the software subcontract activities reviewed with the project manager on both a periodic and event-driven basis? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

The purpose of **Software Quality Assurance (SQA)** 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 involves reviewing and auditing the software products and activities to verify that they comply with the applicable procedures and standards and providing the software project and other appropriate managers with the results of these reviews and audits.

**audit** - An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria. [IEEE-STD-610 Glossary]

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**procedure** - A written description of a course of action to be taken to perform a given task. [IEEE-STD-610 Glossary]

**software quality assurance (SQA)** - (1) A planned and systematic pattern of all actions necessary to provide adequate confidence that a software work product conforms to established technical requirements. (2) A set of activities designed to evaluate the process by which software work products are developed and/or maintained.

**standard** - Mandatory requirements employed and enforced to prescribe a disciplined, uniform approach to software development.

		Yes	No	Does Not Apply	Don't Know
1	Are SQA activities planned? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				
2	Does SQA provide objective verification that software products and activities adhere to applicable standards, procedures, and requirements? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				
3	Are the results of SQA reviews and audits provided to affected groups and individuals (e.g., those who performed the work and those who are responsible for the work)? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				

- |  | Yes                      | No                       | Does Not Apply           | Don't Know               |
|--|--------------------------|--------------------------|--------------------------|--------------------------|
| 4 Are issues of noncompliance that are not resolved within the software project addressed by senior management (e.g., deviations from applicable standards)? .....                 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 5 Does the project follow a written organizational policy for implementing SQA? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 6 Are adequate resources provided for performing SQA activities (e.g., funding and a designated manager who will receive and act on software noncompliance items)? .....           | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 7 Are measurements used to determine the cost and schedule status of the activities performed for SQA (e.g., work completed, effort and funds expended compared to the plan)?..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 8 Are activities for SQA reviewed with senior management on a periodic basis? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |

---

The purpose of **Software Configuration Management (SCM)** is to establish and maintain the integrity of the products of the software project throughout the project's software life cycle. Software Configuration Management involves identifying the configuration of the software (i.e., selected software work products and their descriptions) at given points in time, systematically controlling changes to the configuration, and maintaining the integrity and traceability of the configuration throughout the software life cycle. The work products placed under software configuration management include the software products that are delivered to the customer and the items that are identified with or required to create these software products.

---

**audit** - An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria. [IEEE-STD-610 Glossary]

**configuration item** - An aggregation of hardware, software, or both, that is designated for configuration management and treated as a single entity in the configuration management process. [IEEE-STD-610 Glossary]

**documented procedure** - A written description of a course of action to be taken to perform a given task. [IEEE-STD-610 Glossary]

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**software baseline** - A set of configuration items (software documents and software components) that has been formally reviewed and agreed upon, that thereafter serves as the basis for future development, and that can be changed only through formal change control procedures

**software work product** - Any artifact created as part of defining, maintaining, or using a software process, including process descriptions, plans, procedures, computer programs, and associated documentation, which may or may not be intended for delivery to a customer or end user.

---

		Yes	No	Does Not Apply	Don't Know
1	Are software configuration management activities planned for the project? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				
2	Has the project identified, controlled, and made available the software work products through the use of configuration management? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				

- |   | Yes                      | No                       | Does Not Apply           | Don't Know               |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 3 Does the project follow a documented procedure to control changes to configuration items/units? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 4 Are standard reports on software baselines (e.g., software configuration control board minutes and change request summary and status reports) distributed to affected groups and individuals? | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 5 Does the project follow a written organizational policy for implementing software configuration management activities? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are project personnel trained to perform the software configuration management activities for which they are responsible?.....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Are measurements used to determine the status of activities for software configuration management (e.g., effort and funds expended for software configuration management activities)? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 8 Are periodic audits performed to verify that software baselines conform to the documentation that defines them (e.g., by the SCM group)? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

---

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. Organization Process Focus involves developing and maintaining an understanding of the organization's and projects' software processes and coordinating the activities to assess, develop, maintain, and improve these processes. The organization provides long-term commitments and resources to coordinate the development and maintenance of the software processes across current and future software projects via a group such as a software engineering process group. This group is responsible for the organization's software process activities.

---

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**software process** - A set of activities, methods, practices, and transformations that people use to develop and maintain software and the associated products (e.g., project plans, design documents, code, test cases, and user manuals)

**software process assessment** - 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.

---

	Yes	No	Does Not Apply	Don't Know
1 Are the activities for developing and improving the organization's and project's software processes coordinated across the organization (e.g., via a software engineering process group)? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Is your organization's software process assessed periodically?.....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
3 Does your organization follow a documented plan for developing and improving its software process? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				

- |   | Yes                      | No                       | Does<br>Not<br>Apply     | Don't<br>Know            |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 4 Does senior management sponsor the organization's activities for software process development and improvements (e.g., by establishing long-term plans, and by committing resources and funding)? .....              | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 5 Do one or more individuals have full-time or part-time responsibility for the organization's software process activities (e.g., a software engineering process group)? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are measurements used to determine the status of the activities performed to develop and improve the organization's software process (e.g., effort expended for software process assessment and improvement)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Are the activities performed for developing and improving software processes reviewed periodically with senior management? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

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. Organization Process Definition involves developing and maintaining the organization's standard software process, along with related process assets, such as descriptions of software life cycles, process tailoring guidelines and criteria, the organization's software process database, and a library of software process-related documentation.

**audit** - An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria.

**organization's standard software process** - The operational definition of the basic process that guides the establishment of a common software process across the software projects in an organization. It describes the fundamental software process elements that each software project is expected to incorporate into its defined software process. It also describes the relationships (e.g., ordering and interfaces) between these software process elements.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**software quality assurance (SQA)** - (1) A planned and systematic pattern of all actions necessary to provide adequate confidence that a software work product conforms to established technical requirements. (2) A set of activities designed to evaluate the process by which software work products are developed and/or maintained.

	Yes	No	Does Not Apply	Don't Know
1 Has your organization developed, and does it maintain, a standard software process? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Does the organization collect, review, and make available information related to the use of the organization's standard software process (e.g., estimates and actual data on software size, effort, and cost; productivity data; and quality measurements)? ...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
3 Does the organization follow a written policy for both developing and maintaining its standard software process and related process assets (e.g., descriptions of approved software life cycles)? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				



- |  | Yes                      | No                       | Does<br>Not<br>Apply     | Don't<br>Know            |
|--|--------------------------|--------------------------|--------------------------|--------------------------|
| 4 Do individuals who develop and maintain the organization's standard software process receive the required training to perform these activities? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 5 Are measurements used to determine the status of the activities performed to define and maintain the organization's standard software process (e.g., status of schedule milestones and the cost of process definition activities)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 6 Are the activities and work products for developing and maintaining the organization's standard software process subjected to SQA review and audit? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |

The purpose of the **Training Program** key process area is to develop the skills and knowledge of individuals so they can perform their roles effectively and efficiently. Training Program involves first identifying the training needed by the organization, projects, and individuals, then developing or procuring training to address the identified needs. Some skills are effectively and efficiently imparted through informal vehicles (e.g., on-the-job training and informal mentoring), whereas other skills need more formal training vehicles (e.g., classroom training and guided self-study) to be effectively and efficiently imparted. The appropriate vehicles are selected and used.

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

		Yes	No	Does Not Apply	Don't Know
1	Are training activities planned? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				
2	Is training <u>provided</u> for developing the skills and knowledge needed to perform software managerial and technical roles? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				
3	Do members of the software engineering group and other software-related groups <u>receive</u> the training necessary to perform their roles? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				
4	Does your organization follow a written organizational policy to meet its training needs?.....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Comments:				

- |  | Yes                      | No                       | Does<br>Not<br>Apply     | Don't<br>Know            |
|--|--------------------------|--------------------------|--------------------------|--------------------------|
| 5 Are adequate resources provided to implement the organization's training program (e.g., funding, software tools, appropriate training facilities)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 6 Are measurements used to determine the quality of the training program? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 7 Are training program activities reviewed with senior management on a periodic basis?.....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |

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. Integrated Software Management involves developing the project's defined software process and managing the software project using this defined software process. The project's defined software process is tailored from the organization's standard software process to address the specific characteristics of the project. The software development plan is based on the project's defined software process and describes how the activities of the project's defined software process will be implemented and managed.

**audit** - An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria.

**organization's standard software process** - The operational definition of the basic process that guides the establishment of a common software process across the software projects in an organization. It describes the fundamental software process elements that each software project is expected to incorporate into its defined software process. It also describes the relationships (e.g., ordering and interfaces) between these software process elements.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**project's defined software process** - The operational definition of the software process used by a project. The project's defined software process is a well-characterized and understood software process, described in terms of software standards, procedures, tools, and methods. It is developed by tailoring the organization's standard software process to fit the specific characteristics of the project.

**software quality assurance (SQA)** - (1) A planned and systematic pattern of all actions necessary to provide adequate confidence that a software work product conforms to established technical requirements. (2) A set of activities designed to evaluate the process by which software work products are developed and/or maintained.

**tailoring** - To modify a process, standard, or procedure to better match process or product requirements.

		Yes	No	Does Not Apply	Don't Know
1	Was the <u>project's</u> defined software process developed by tailoring the <u>organization's</u> standard software process? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:					
2	Is the project planned and managed in accordance with the project's defined software process? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:					

- |  | Yes                      | No                       | Does<br>Not<br>Apply     | Don't<br>Know            |
|--|--------------------------|--------------------------|--------------------------|--------------------------|
| <p>3 Does the project follow a written organizational policy requiring that the software project be planned and managed using the organization's standard software process? .....</p> <p>Comments:</p> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <p>4 Is training required for individuals tasked to tailor the organization's standard software process to define a software process for a new project?.....</p> <p>Comments:</p>                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <p>5 Are measurements used to determine the effectiveness of the integrated software management activities (e.g., frequency, causes and magnitude of replanning efforts)? .....</p> <p>Comments:</p>   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <p>6 Are the activities and work products used to manage the software project subjected to SQA review and audit? .....</p> <p>Comments:</p>  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

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 involves performing the engineering tasks to build and maintain the software using the project's defined software process and appropriate methods and tools. The software engineering tasks include analyzing the system requirements allocated to software, developing the software architecture, designing the software, implementing the software in the code, and testing the software to verify that it satisfies the specified requirements.

**audit** - An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**project's defined software process** - The operational definition of the software process used by a project. The project's defined software process is a well-characterized and understood software process, described in terms of software standards, procedures, tools, and methods. It is developed by tailoring the organization's standard software process to fit the specific characteristics of the project.

**software quality assurance (SQA)** - (1) A planned and systematic pattern of all actions necessary to provide adequate confidence that a software work product conforms to established technical requirements. (2) A set of activities designed to evaluate the process by which software work products are developed and/or maintained.

**software work product** - Any artifact created as part of defining, maintaining, or using a software process, including process descriptions, plans, procedures, computer programs, and associated documentation, which may or may not be intended for delivery to a customer or end user.

		Yes	No	Does Not Apply	Don't Know
1	Are the software work products produced according to the project's defined software process? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:					
2	Is consistency maintained across software work products (e.g., is the documentation tracing allocated requirements through software requirements, design, code, and test cases maintained)? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:					

- |  | Yes                      | No                       | Does<br>Not<br>Apply     | Don't<br>Know            |
|--|--------------------------|--------------------------|--------------------------|--------------------------|
| <p>3 Does the project follow a written organizational policy for performing the software engineering activities (e.g., a policy which requires the use of appropriate methods and tools for building and maintaining software products)? .....</p> <p>Comments:</p>        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <p>4 Are adequate resources provided for performing the software engineering tasks (e.g., funding, skilled individuals, and appropriate tools)? .....</p> <p>Comments:</p>   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <p>5 Are measurements used to determine the functionality and quality of the software products (e.g., numbers, types, and severity of defects identified)? .....</p> <p>Comments:</p>  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <p>6 Are the activities and work products for engineering software subjected to SQA reviews and audits (e.g., is required testing performed, are allocated requirements traced through the software requirements, design, code and test cases)? .....</p> <p>Comments:</p> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

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 involves the software engineering group's participation with other project engineering groups to address system-level requirements, objectives, and issues. Representatives of the project's engineering groups participate in establishing the system-level requirements, objectives, and plans by working with the customer and end users, as appropriate. These requirements, objectives, and plans become the basis for all engineering activities.

**commitment** - A pact that is freely assumed, visible, and expected to be kept by all parties.

**event-driven review/activity** - A review or activity that is performed based on the occurrence of an event within the project (e.g., a formal review or the completion of a life-cycle stage).

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

	Yes	No	Does Not Apply	Don't Know
1 On the project, do the software engineering group and other engineering groups collaborate with the customer to establish the system requirements? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Do the engineering groups agree to the commitments as represented in the overall project plan? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
3 Do the engineering groups identify, track, and resolve intergroup issues (e.g., incompatible schedules, technical risks, or system-level problems)? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				



- |   | Yes                      | No                       | Does Not Apply           | Don't Know               |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 4 Is there a written organizational policy that guides the establishment of interdisciplinary engineering teams? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 5 Do the support tools used by different engineering groups enable effective communication and coordination (e.g., compatible word processing systems, database systems, and problem tracking systems)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are measures used to determine the status of the intergroup coordination activities (e.g., effort expended by the software engineering group to support other groups)? .....                                | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Are the activities for intergroup coordination reviewed with the project manager on both a periodic and event-driven basis? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

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 defects that might be prevented. Peer Reviews involve a methodical examination of software work products by the producers' peers to identify defects and areas where changes are needed. The specific products that will undergo a peer review are identified in the project's defined software process and scheduled as part of the software project planning activities.

**audit** - An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria.

**peer review** - A review of a software work product, following defined procedures, by peers of the producers of the product for the purpose of identifying defects and improvements.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**software quality assurance (SQA)** - (1) A planned and systematic pattern of all actions necessary to provide adequate confidence that a software work product conforms to established technical requirements. (2) A set of activities designed to evaluate the process by which software work products are developed and/or maintained.

**software work product** - Any artifact created as part of defining, maintaining, or using a software process, including process descriptions, plans, procedures, computer programs, and associated documentation, which may or may not be intended for delivery to a customer or end user.

		Yes	No	Does Not Apply	Don't Know
1	Are peer reviews planned? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:					
2	Are actions associated with defects that are identified during peer reviews tracked until they are resolved?.....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:					
3	Does the project follow a written organizational policy for performing peer reviews? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:					

- |  | Yes                      | No                       | Does<br>Not<br>Apply     | Don't<br>Know            |
|--|--------------------------|--------------------------|--------------------------|--------------------------|
| 4 Do participants of peer reviews receive the training required to perform their roles?.....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 5 Are measurements used to determine the status of peer review activities (e.g., number of peer reviews performed, effort expended on peer reviews, and number of work products reviewed compared to the plan)?..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 6 Are peer review activities and work products subjected to SQA review and audit (e.g., planned reviews are conducted and follow-up actions are tracked)? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |

The purpose of **Quantitative Process Management** is to control the process performance of the software project quantitatively. Quantitative Process Management involves taking measurements of the process performance, analyzing these measurements, and making adjustments to maintain process performance within acceptable limits. When the process performance is stabilized within acceptable limits, the project's defined software process, the associated measurements, and the acceptable limits for the measurements are established as a baseline and used to control process performance quantitatively.

**event-driven review/activity** - A review or activity that is performed based on the occurrence of an event within the project (e.g., a formal review or the completion of a life-cycle stage).

**organization's standard software process** - The operational definition of the basic process that guides the establishment of a common software process across the software projects in an organization. It describes the fundamental software process elements that each software project is expected to incorporate into its defined software process. It also describes the relationships (e.g., ordering and interfaces) between these software process elements.

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**process capability** - The range of expected results that can be achieved by following a process.

**project's defined software process** - The operational definition of the software process used by a project. The project's defined software process is a well-characterized and understood software process, described in terms of software standards, procedures, tools, and methods. It is developed by tailoring the organization's standard software process to fit the specific characteristics of the project.

	Yes	No	Does Not Apply	Don't Know
1 Does the project follow a documented plan for conducting quantitative process management? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Is the performance of the <u>project's</u> defined software process controlled quantitatively (e.g., through the use of quantitative analytic methods)? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				

- |   | Yes                      | No                       | Does Not Apply           | Don't Know               |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 3 Is the process capability of the <u>organization's</u> standard software process known in quantitative terms? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 4 Does the project follow a written organizational policy for measuring and controlling the performance of the project's defined software process (e.g., projects plan for how to identify, analyze, and control special causes of variations)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 5 Are adequate resources provided for quantitative process management activities (e.g., funding, software support tools, and organizational measurement program)? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are measurements used to determine the status of the quantitative process management activities (e.g., cost of quantitative process management activities and accomplishment of milestones for quantitative process management activities)? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Are the activities for quantitative process management reviewed with the project manager on both a periodic and event-driven basis?.....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

---

**Software Quality Management** involves defining quality goals for the software products, establishing plans to achieve these goals, and monitoring and adjusting the software plans, software work products, activities, and quality goals to satisfy the needs and desires of the customer and end user. Quantitative product quality goals are established based on the needs of the organization, customer, and end user for high-quality products. So that these goals may be achieved, the organization establishes strategies and plans, and the project specifically adjusts its defined software process, to accomplish the quality goals.

---

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

---

	Yes	No	Does Not Apply	Don't Know
1 Are the activities for managing software quality planned for the project? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Does the project use measurable and prioritized goals for managing the quality of its software products (e.g., functionality, reliability, maintainability and usability)? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
3 Are measurements of quality compared to goals for software product quality to determine if the quality goals are satisfied? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
4 Does the project follow a written organizational policy for managing software quality? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				

- |  | Yes                      | No                       | Does<br>Not<br>Apply     | Don't<br>Know            |
|--|--------------------------|--------------------------|--------------------------|--------------------------|
| 5 Do members of the software engineering group and other software-related groups receive required training in software quality management (e.g., training in collecting measurement data and benefits of quantitatively managing product quality)? ... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 6 Are measurements used to determine the status of the activities for managing software quality (e.g., the cost of poor quality)? ....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |
| 7 Are the activities performed for software quality management reviewed with senior management on a periodic basis? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:  |                          |                          |                          |                          |

---

**Defect Prevention** involves analyzing defects that were encountered in the past and taking specific actions to prevent the occurrence of those types of defects in the future. The defects may have been identified on other projects as well as in earlier stages or tasks of the current project. Trends are analyzed to track the types of defects that have been encountered and to identify defects that are likely to recur. Both the project and the organization take specific actions to prevent recurrence of the defects.

---

**audit** - An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria.

**causal analysis meeting** - A meeting, conducted after completing a specific task, to analyze defects uncovered during the performance of that task.

**common cause (of a defect)** - A cause of a defect that is inherently part of a process or system. Common causes affect every outcome of the process and everyone working in the process.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**software quality assurance (SQA)** - (1) A planned and systematic pattern of all actions necessary to provide adequate confidence that a software work product conforms to established technical requirements. (2) A set of activities designed to evaluate the process by which software work products are developed and/or maintained.

---

	Yes	No	Does Not Apply	Don't Know
1 Are defect prevention activities planned? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Does the project conduct causal analysis meetings to identify common causes of defects? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
3 Once identified, are common causes of defects prioritized and systematically eliminated? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				



- |   | Yes                      | No                       | Does Not Apply           | Don't Know               |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 4 Does the project follow a written organizational policy for defect prevention activities? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 5 Do members of the software engineering group and other software-related groups receive required training to perform their defect prevention activities (e.g., training in defect prevention methods and the conduct of task kick-off or causal analysis meetings)?..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are measurements used to determine the status of defect prevention activities (e.g., the time and cost for identifying and correcting defects and the number of action items proposed, open, and completed)? .....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Are the activities and work products for defect prevention subjected to SQA review and audit?.....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

---

**Technology Change Management** involves identifying, selecting, and evaluating new technologies, and incorporating effective technologies into the organization. The objective is to improve software quality, increase productivity, and decrease the cycle time for product development. The organization establishes a group (such as a software engineering process group or a technology support group) that works with the software projects to introduce and evaluate new technologies and manage changes to existing technologies. Particular emphasis is placed on technology changes that are likely to improve the capability of the organization's standard software process. Pilot efforts are performed to assess new and unproven technologies before they are incorporated into normal practice. With appropriate sponsorship of the organization's management, the selected technologies are incorporated into the organization's standard software process and current projects, as appropriate.

---

**documented procedure** - A written description of a course of action to be taken to perform a given task.  
 [IEEE-STD-610 Glossary]

**organization's standard software process** - The operational definition of the basic process that guides the establishment of a common software process across the software projects in an organization. It describes the fundamental software process elements that each software project is expected to incorporate into its defined software process. It also describes the relationships (e.g., ordering and interfaces) between these software process elements.

**periodic review/activity** - A review or activity that occurs at specified regular time intervals.

---

	Yes	No	Does Not Apply	Don't Know
1 Does the organization follow a plan for managing technology changes?.....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Are new technologies evaluated to determine their effect on quality and productivity? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
3 Does the organization follow a documented procedure for incorporating new technologies into the <u>organization's</u> standard software process? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				

- |   | Yes                      | No                       | Does<br>Not<br>Apply     | Don't<br>Know            |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 4 Does senior management sponsor the organization's activities for managing technology change (e.g., by establishing long-term plans and commitments for funding, staffing, and other resources)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 5 Do process data exist to assist in the selection of new technology? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are measurements used to determine the status of the organization's activities for managing technology change (e.g., the effect of implementing technology changes)? .....                            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Are the organization's activities for managing technology change reviewed with senior management on a periodic basis?.....  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

---

**Process Change Management** involves defining process improvement goals and, with senior management sponsorship, proactively and systematically identifying, evaluating, and implementing improvements to the organization's standard software process and the projects' defined software processes on a continuous basis. Training and incentive programs are established to enable and encourage everyone in the organization to participate in process improvement activities. Improvement opportunities are identified and evaluated for potential payback to the organization. Pilot efforts are performed to assess process changes before they are incorporated into normal practice. When software process improvements are approved for normal practice, the organization's standard software process and the projects' defined software processes are revised as appropriate.

---

**documented procedure** - A written description of a course of action to be taken to perform a given task.  
 [IEEE-STD-610 Glossary]

**organization's standard software process** - The operational definition of the basic process that guides the establishment of a common software process across the software projects in an organization. It describes the fundamental software process elements that each software project is expected to incorporate into its defined software process. It also describes the relationships (e.g., ordering and interfaces) between these software process elements.

**periodic review/activity** - A review/activity that occurs at a specified regular time interval, rather than at the completion of major events.

**policy** - A guiding principle, typically established by senior management, which is adopted by an organization or project to influence and determine decisions.

**project's defined software process** - The operational definition of the software process used by a project. The project's defined software process is a well-characterized and understood software process, described in terms of software standards, procedures, tools, and methods. It is developed by tailoring the organization's standard software process to fit the specific characteristics of the project.

---

	Yes	No	Does Not Apply	Don't Know
1 Does the organization follow a documented procedure for developing and maintaining plans for software process improvement? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				
2 Do people throughout your organization participate in software process improvement activities (e.g., on teams to develop software process improvements)? .....	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:				

- |   | Yes                      | No                       | Does Not Apply           | Don't Know               |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 3 Are improvements continually made to the organization's standard software process and the projects' defined software processes? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 4 Does the organization follow a written policy for implementing software process improvements? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 5 Is training in software process improvement required for both management and technical staff? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 6 Are measurements made to determine the status of the activities for software process improvement (e.g., the effect of implementing each process improvement compared to its defined goals)? ..... | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |
| 7 Are software process improvement efforts reviewed with senior management on a periodic basis? .....   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments:   |                          |                          |                          |                          |

**Thank you very much for your time and effort!!!**



## **Software Process Maturity Questionnaire**

Software Engineering Institute  
Carnegie Mellon University  
Pittsburgh, PA 15213-3890

# Instruction Placard - Software Process Maturity Questionnaire

## Instructions

1 To the right of each question there are boxes for the four possible responses: **Yes**, **No**, **Does Not Apply**, and **Don't Know**.

Check **Yes** when:

- The practice is well established and consistently performed.
  - The practice should be performed nearly always in order to be considered well-established and consistently performed as a standard operating procedure.

Check **No** when:

- The practice is not well established or is inconsistently performed.
  - The practice may be performed sometimes, or even frequently, but it is omitted under difficult circumstances.

Check **Does Not Apply** when:

- You have the required knowledge about your project or organization and the question asked, but you feel the question does not apply for your project.
  - For example, the entire section on “Software Subcontract Management” may not apply to your project if you don't work with any subcontractors.

Check **Don't Know** when:

- You are uncertain about how to answer the question.

2 Use the **Comments** spaces for any elaborations or qualifications about your answers to the questions.

3 Check one of the boxes for each question. Please answer all of the questions.

## Organizational Terms

**affected groups** - Groups with related responsibilities or obligations whose work performance might be impacted. Such groups might include software engineering, software estimating, system engineering, hardware engineering, system test, software quality assurance, software configuration management, finance, documentation support, and software engineering process.

**groups external to the organization** - Groups outside of the organizational unit being assessed. Such groups might include customers and end users.

**organization** - A unit within a company or other entity (e.g., government agency or branch of service) within which many projects are managed as a whole. All projects within an organization share a common top-level manager and common policies.

**project** - An undertaking requiring concerted effort, which is focused on developing and/or maintaining a specific product. The product may include hardware, software, and other components. Typically a project has its own funding, cost accounting, and delivery schedule.

**project manager** - The role with total business responsibility for an entire project; the individual who directs, controls, administers, and regulates a project building a software or hardware/software system. The project manager is the individual ultimately responsible to the customer.

**senior manager** - A management role at a high enough level in an organization that the primary focus is the long-term vitality of the organization, rather than short-term project and contractual concerns and pressures. In general, a senior manager for engineering would have responsibility for multiple projects.

**software engineering group** - The collection of individuals (both managers and technical staff) who have responsibility for software development and maintenance activities (i.e., requirements analysis, design, code, and test) for a project. Groups performing software-related work, such as the software quality assurance group, the software configuration management group, and the software engineering process group, are not included in the software engineering group.

**software engineering process group (SEPG)** - A group of specialists who facilitate the definition, maintenance, and improvement of the software process used by the organization. In the key practices, this group is generically referred to as “the group responsible for the organization’s software process activities.”

**software engineering staff** - The software technical people (e.g., analysts, programmers, and engineers), including software task leaders, who perform the software development and maintenance activities for the project, but who are not managers.

**software manager** - Any manager, at a project or organizational level, who has direct responsibility for software development and/or maintenance.

**subcontractor** - an individual, partnership, corporation, or association that contracts with an organization (i.e., the prime contractor) to design, develop, and/or manufacture one or more products.

**system engineering group** - The collection of departments, managers, and staff who have responsibility for specifying the system requirements, allocating the system requirements to the hardware and software components, specifying the interfaces between the hardware and software components, and monitoring the design and development of these components to ensure conformance with their specifications.



## Change Request - Software Process Maturity Questionnaire

---

**Product :** Maturity Questionnaire

**Version:** 1.1.0

**SEI Assigned Tracking Number:** \_\_\_\_\_

---

**Name of Submitting Organization:** \_\_\_\_\_

**Organization Contact:** \_\_\_\_\_ **Telephone:** \_\_\_\_\_

**Mailing Address:**

---

**Date:** \_\_\_\_\_ **Short Title:** \_\_\_\_\_

**Change Location Tag:** \_\_\_\_\_  
(use Key Process Area, Page #, Question #)

**Proposed Change:**

**Rationale for Change:**

---

**Note:** For the SEI to take appropriate action on a change request, we must have a clear description of the recommended change, along with a supporting rationale.

**Send U.S. mail to:** Empirical Methods-MQ Change Requests, Software Process Program, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890.

**Send Packages to:** Empirical Methods-MQ Change Requests, Software Process Program, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA 15213-2691.