SMC-TR-04-01

AEROSPACE REPORT NO.
TR-2003(8550)-1

A Practical Guidebook for Performing Software Capability Appraisals

20 September 2003

Prepared by

K. L. OWENS
Software Engineering Subdivision
Computer Systems Division

B. R. TROUP
Space-Based Surveillance Division

Prepared for

SPACE AND MISSILE SYSTEMS CENTER
AIR FORCE SPACE COMMAND
2430 E. El Segundo Boulevard
Los Angeles Air Force Base, CA 90245

20031117 049

Engineering and Technology Group

THE AEROSPACE CORPORATION
El Segundo, California

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED

This report was submitted by The Aerospace Corporation, El Segundo, CA 90245-4691, under Contract No. F04701-00-C-0009 with the Space and Missile Systems Center, 2430 E. El Segundo Blvd., Los Angeles Air Force Base, CA 90245. It was reviewed and approved for The Aerospace Corporation by Mary A. Rich, Principal Director, Software Engineering Subdivision. Michael Zambrana was the project officer for the Mission-Oriented Investigation and Experimentation (MOIE) program.

This report has been reviewed by the Public Affairs Office (PAS) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication. Publication of this report does not constitute Air Force approval of the report's findings or conclusions. It is published only for the exchange and stimulation of ideas.

Michael Zambrana
SMC/AXE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (*DD-MM-YYYY*)	2. REPORT TYPE	3. DATES COVERED (*From - To*)
20-09-2003		

4. TITLE AND SUBTITLE

A Practical Guidebook for Performing Software Capability Appraisals

5a. CONTRACT NUMBER
F04701-00-C-0009

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

K. L. Owens and B. R. Troup

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

The Aerospace Corporation
Laboratory Operations
El Segundo, CA 90245-4691

8. PERFORMING ORGANIZATION REPORT NUMBER

TR-2003(8550)-1

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Space and Missile Systems Center
Air Force Space Command
2450 E. El Segundo Blvd.
Los Angeles Air Force Base, CA 90245

10. SPONSOR/MONITOR'S ACRONYM(S)
SMC

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
SMC-TR-04-01

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This guidebook provides practical guidance on why software capability appraisals are performed and how to prepare for them. It incorporates lessons learned from past and recent Software Development Capability Evaluation (SDCE) appraisals and provides example instructions and an initial set of questions to be tailored for a specific program. Also provided are new questions and criteria for evaluating the contractor's processes for developing COTS software-based systems.

15. SUBJECT TERMS

Acquisition, Appraisal, Assessment, Capability assessment, Capability evaluation, Capability Maturity Model (CMM), Capability Maturity Model Integration (CMMI), Capability, Contractor, COTS, Evaluation, Guidebook, Software, Software capability appraisal, Software Development Capability Evaluation, SDCE

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
					Karen Owens
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (*include area code*)
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED		151	(310)336-5909

Acknowledgments

This work would not have been possible without funding and assistance from the Office of the Chief Engineer and the Mission-Oriented Investigation and Experimentation (MOIE) Program (Software Acquisition Task)

This document contains blank pages that were not filmed.

Preface

Capability Maturity Model®, CMM®, Capability Maturity Model for Software®, SW-CMM®, Capability Maturity Model Integration®, and CMMI® are registered trademarks of the Software Engineering Institute (SEI) and Carnegie Mellon University (CMU). CMM-Based Assessment for Internal Process Improvement℠, CBA-IPI℠, Standard CMMI Appraisal Method for Process Improvement℠, SCAMPI℠, Software Capability Evaluation℠, and SCE℠ are service marks of the SEI and CMU. The registered trademark and service mark symbols will not be used in the rest of the document.

Contents

Part I. PERFORMING SOFTWARE APPRAISALS **1**

1. Introduction 1

1.1 Purpose 1

1.2 Definitions 1

1.3 Document Contents 2

2. Appraisal Methods 3

2.1 Process Models and Appraisal Methods 4

2.2 Appraisal Process and Phases 5

2.2.1 Plan and Prepare for an Appraisal 6

2.2.2 Conduct Appraisal 7

2.2.3 Report Appraisal Results 8

3. Performing an Evaluation 9

3.1 Considerations for Performing an Evaluation 9

3.1.1 Determine Evaluation Need 9

3.1.2 Determine Evaluation Objectives 9

3.2 Evaluation Constraints 11

3.2.1 Cost Constraints 11

3.2.2 Schedule Constraints 12

3.3 Evaluation Scope, Planning, and Timing 12

3.3.1 Determine Evaluation Scope 13

3.3.2 Develop Evaluation Plan 13

3.3.3 Evaluation Timing 14

3.4 Frequently Asked Questions Regarding Appraisals 15

Part II. USING THE SDCE AS AN APPRAISAL TECHNIQUE **19**

1. Background on the SDCE 19

2. Software Development Capability Evaluation Model Structure 21
2.1 The Functional Areas 21
2.2 The Critical Capability Areas (CCAs) 22
2.3 The Critical Capabilities, Questions and Criteria 23
3. Software Development Capability Evaluation Process 29
3.1 Plan and Prepare for an SDCE 30
3.1.1 Determine Program Risks and Resources 30
3.1.2 Define Processes 30
3.1.3 Prepare Plan and Schedule 30
3.1.4 Tailor SDCE (determine questions and criteria) 31
3.1.5 Incorporate into RFP 31
3.1.6 Select and Prepare Team 34
3.2 Conduct Evaluation 35
3.2.1 Review Proposals/Offeror Responses to Questionnaire 35
3.2.2 Prepare Evaluation Notices (ENs) 37
3.2.3 Perform Site Visits (optional) 37
3.2.4 Analyze EN Responses 38
3.2.5 Establish SDCE Results (Determine strengths, weaknesses, and risks) 38
3.2.6 Integrate with Source Selection 38
3.3 Report Evaluation Results 39
3.3.1 Conduct Feedback (optional) 39
3.3.2 Transition SDCE results 40
3.3.3 Program follow-through 40
4. The Basic Software Development Capability Evaluation 41
4.1 Background 41
4.2 Tailoring/Augmenting the Basic SDCE 42
4.3 Resources Required 43
4.4 Benefits of the Basic SDCE 44
4.5 Disadvantages of the Basic SDCE 45

5. The Large Software Development Capability Evaluation 47
5.1 Background 47
5.2 The Large SDCE Pilot 47
5.2.1 Description 47
5.2.2 Questions and Criteria 47
5.2.3 Contractors' Instructions for Completing the Level 3-Equivalent SDCE 48
5.3 Evaluation Team Training 48
5.3.1 Lessons Learned 49
5.4 Evaluation Process 49
5.4.1 Teaming Arrangements 49
5.4.2 Consensus Building 50
5.4.3 Roll-Up Process 51
5.4.4 Resource Usage 52
5.4.5 Lessons Learned 53
Appendix A—Basic SDCE 55
Appendix B—Instructions for the Large SDCE 73
Appendix C— CMM Level 3-Equivalent Core Set of Questions and Criteria 97
Appendix D—Commercial Off-The-Shelf Software Questions and Criteria 129
Appendix E—Bibliography 135
Appendix F—Acronyms and Definitions 139

Figures

Part I

1-1. Appraisals: assessments vs. evaluations 3
2-2. Appraisal phases 5
3-1. The Use of SCAMPI in the source selection process. 15

Part II

2-1. Example of the SDCE model structure hierarchy. 21

2-2. SDCE Model—Top Level. 22

2-3. Critical capabilities for Functional Area 1 24

2-4. Critical capabilities for Functional Area 2 24

2-5. Critical capabilities for Functional Area 3 25

2-6. Critical capabilities for Functional Area 4 25

2-7. Critical capabilities for Functional Area 5 26

2-8. Critical capabilities for Functional Area 6 (Top Level). 26

2-9. Critical capabilities for Functional Area 6 (Part I). 27

2-10. Critical capabilities for Functional Area 6 (Part II) 27

3-2. Sample language for Section L—Instructions to Offerors. 32

3-3. Sample language for Section M—Evaluation Criteria. 32

3-4. Recommended placement of SDCE in RFP structure. 33

3-5. Poor placement of SDCE in RFP structure 33

3-6. SDCE outside of source selection (SS) 34

3-7. Cover sheet for SDCE evidence of use. 36

3-8. Sample completed cover sheet for SDCE evidence of use. 36

5-1. SDCE team consensus process 50

5-2. SDCE resources and duration (AF & Aerospace). 53

Tables

Part I

2-1. Process Models and Their Related Appraisal Methods 4

3-1. Method Objectives for Evaluation During Source Selection 10

3-2. Cost Constraint Mitigations, Savings, and Risks 11

Part II

4-1. SDCE Statistics for SMC and NRO 41

5-1. Issue Bins and Categories 52

Part I. PERFORMING SOFTWARE APPRAISALS

1. Introduction

1.1 Purpose

The purpose of this guidebook is to provide the reader with the guidance for performing a software capability appraisal. Its focus is on software process appraisals, which require an understanding of good software development methods and standard software products. The guidebook goal is to provide readers with some understanding of each appraisal method, the reasons for selecting a particular method, and the process for applying the method to appraise a particular program. Specific guidance information is provided for using the Air Force's Software Development Capability Evaluation (SDCE). This document does not provide guidance in software development methods or software product development.

This guidebook includes guidance for both small and large appraisals. It presents guidance for performing an evaluation during source selection. It also includes guidance for performing an evaluation during a contract performance period, e.g., to support a subsequent downselection, to identify candidate software-related risk areas requiring additional insight during contract performance, or to form the basis for process improvement by the contractors.

1.2 Definitions

Although the words "**appraisal**," "**assessment**," and "**evaluation**" are commonly used as synonyms in the English language, these words have very distinct meanings when used in the software process discipline, as defined below.

An "**appraisal**" is a systematic method that employs a defined model for examining an organization's (e.g., contractor's or contractor project team's) development and maintenance processes. "**Appraisal**" is the umbrella term that includes both assessments and evaluations.

An "**assessment**" is an examination with respect to a reference model, performed internally by an organization for itself for the purpose of process improvement.

An "**evaluation**" is an examination with respect to a reference model, performed on an organization by an external entity (e.g., by a Government team on a contractor team or by a prime contractor on a subcontractor).

1.3 Document Contents

Part I contains the basic definitions used in a software capability appraisal, the general appraisal process, and considerations for performing a software capability appraisal. We also explain the differences between evaluations and assessments, where the evaluation and assessment methods are described at a top level with references to more detailed literature for further investigation by the reader.

Specific example information is provided in Part II of this document using the context of the Air Force's Software Development Capability Evaluation (SDCE). This portion of the guidebook also includes discussions of the Basic SDCE, a small questionnaire that can be used by an SDCE team as a starting point for developing a program-specific set of questions and criteria, and the Large SDCE, also referred to as the Acquisition Category I (ACAT I) or Level 3-equivalent SDCE.

Appendix A provides the Basic SDCE instructions and questionnaire. Appendix B provides the Level 3-equivalent SDCE instructions. Appendix C provides the Level 3-equivalent set of questions and criteria and their mappings to the Capability Maturity Model Software (SW-CMM) Key Process Areas (KPAs). Appendix D provides a set of questions and criteria specific to Commercial off-the-shelf (COTS) software processes (e.g., selection, application, and maintenance). Appendix E lists the references noted in the guidebook. Appendix F defines the acronyms used.

2. Appraisal Methods

An appraisal follows a systematic method that employs a defined reference model for examining an organization's development processes. In an appraisal, the organization's or project's processes, policies, practices, and procedures are examined with respect to the reference model to determine whether the processes meet the model; to identify strengths, risks, and inadequacies or weaknesses; and to identify gaps or improvement opportunities. Using the defined reference model, substantiating information is gathered from current or recent projects to determine the process capability of the organization. "Appraisal" is an umbrella term that covers both assessments and evaluations.

Figure 2-1 shows the important differences between assessments (done for internal process improvement) and evaluations (done to examine another organization's capability). Assessments focus on the organization while evaluations focus on a specific program.

The purposes of assessments and evaluations are related, but differ in their respective applications. The primary purpose of an **assessment** is to provide results that support senior management decision making, e.g., where to allocate scarce resources, for disciplined process improvement. The assess-

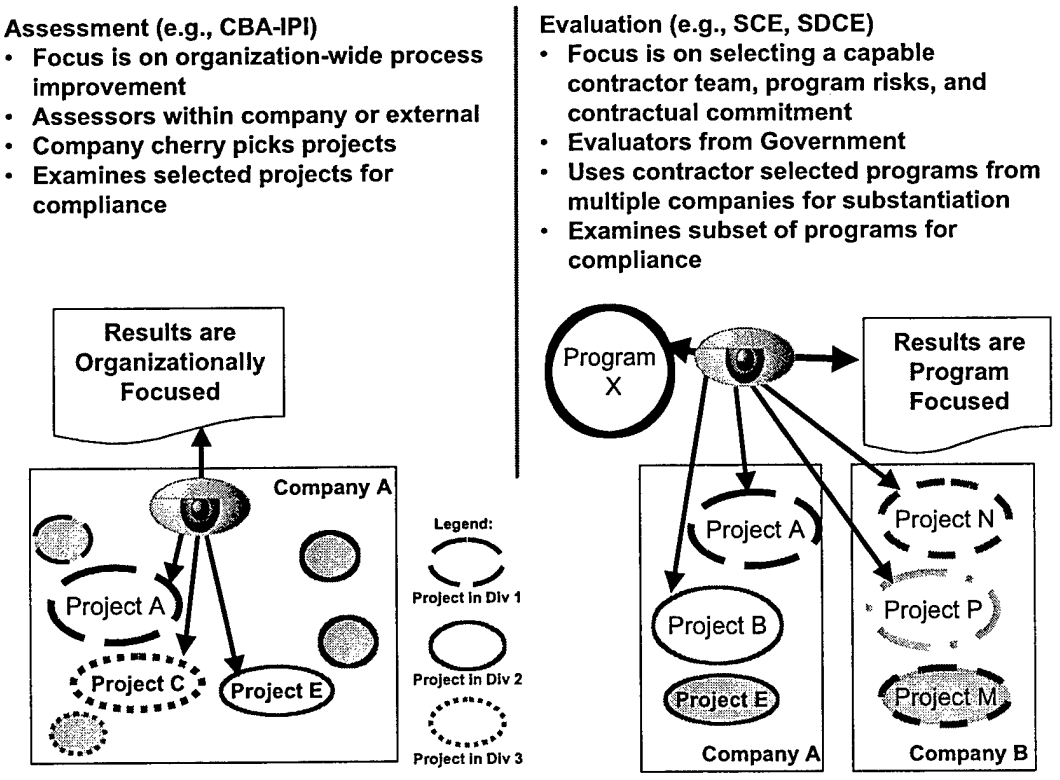


Figure 2-1. Appraisals: assessments vs. evaluations.

ment does this systematically by obtaining results relative to a reference model on an organization's existing process strengths, weaknesses, improvement opportunities, and risks.

On the other hand, the primary purpose of a Government **evaluation** for source selection or contract performance monitoring is to elicit information on a contractor's or contractor team's process capability relative to a particular project or program. The primary objective of the use of a Government evaluation for source selection is to increase the likelihood of selecting a contractor capable of developing the required software within the program constraints, while also identifying risks inherent in the contractor team's development processes and proposed development approach.

The primary objective of the use of a Government evaluation for contract performance monitoring is to identify process weaknesses and risks to both the contractor and the Government, which should stimulate process improvement and risk mitigation by the contractor team.

In addition, a contractor may perform evaluations on its subcontractors to determine whether their processes are compatible or at the same maturity level. In this case, areas of weakness are noted, and subcontractors may be required to improve their process performance or else risk the loss of the subcontract or award fees.

2.1 Process Models and Appraisal Methods

Each appraisal method examines a set of project and organizational processes with respect to its associated process model. Table 2-1 relates these appraisal methods to their respective models. The Software Engineering Institute's models and appraisal methods are included because they are well known and widely used in commercial software development organizations, the aerospace industry, and the Government. The SDCE has been the evaluation model for the Air Force Materiel Command's (AFMC's) Aeronautical Systems Center (ASC) and Air Force Space Command's Space and Missile Systems Center (SMC) since the inception of the SDCE in 1994. As shown in Table 2-1, assessments are performed using the CMM-Based Appraisal for Internal Process Improvement (CBA-IPI) method with the Capability Maturity Model for Software (SW-CMM), or the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) with the Capability Maturity Model Integration (CMMI) models. For evaluations, the SDCE method is used with the SDCE model, the Software Capability Evaluation (SCE) method is used with the SW-CMM, and the SCAMPI is used with the CMMI model. Maintenance of the SW-CMM model and the CBA-IPI and SCE methods has stopped. Training on the model and its appraisal methods will be phased out by January 2004 and is being replaced by training on the CMMI model and its associated method (i.e., SCAMPI).

Table 2-1. Process Models and Their Related Appraisal Methods

Developing Organization	Model	Appraisal Method	
		Evaluation	Assessment
AFMC	SDCE	SDCE	N/A
SEI	SW-CMM	SCE	CBA-IPI
SEI	CMMI	SCAMPI	SCAMPI

The SW-CMM and the CMMI are each models for organizational process management and quality improvement developed by the Software Engineering Institute (SEI) with military, industry, and Federally Funded Research and Development Center (FFRDC) participation. The SW-CMM covers only software engineering and management, while the CMMI can cover software engineering and management, systems engineering and management, integrated product and process development, and supplier sourcing. This guidebook does not cover the Software Engineering Institute's models and appraisal methods in detail; however, references for these models and methods are provided in Appendix E.

A team of military, industry, and FFRDC organizations also developed the SDCE model and method. The primary objective of the SDCE is to evaluate the processes being proposed for a particular program, generally for a Government source selection. The questions and criteria that make up the SDCE model cover program management and systems engineering as related to the software development process, software development itself, and integral processes required for good software practice (e.g., peer reviews, software quality assurance). Additional questions and criteria have been added under program-specific technologies, such as object-oriented development and distributed processing, to keep the SDCE current with emerging technologies. The structure of the SDCE model and the use of the SDCE process are described in detail in Part II of this document.

SDCE evaluations consider the quality of the contractor's defined or proposed software processes for the particular project or program and determine whether these processes have been used successfully on the same or similar software projects by the organization being evaluated. The SCE and SCAMPI evaluations examine the organization's processes used on three or more of its projects with respect to the criteria in the reference model and determine whether the selected projects are correctly following those processes. In general, the project or program under consideration is not one of the several evaluated during the SCE or SCAMPI. The assumption is that the evaluated organization will use their organization-wide processes on every project under its purview, including the project of interest to the evaluators. While a SDCE evaluation looks at the contractor team, the SCE and SCAMPI evaluations generally look at an individual contractor organization.

2.2 Appraisal Process and Phases

All appraisal methods described in Section 2.1 follow the same basic process, shown in Figure 2-2. The three boxes in the figure represent the three phases of the appraisal process: planning and preparation for an appraisal, conducting the appraisal, and reporting appraisal results. Prior to conducting the appraisal, the appraisal team leader provides assistance to the sponsoring organization (e.g., pro-

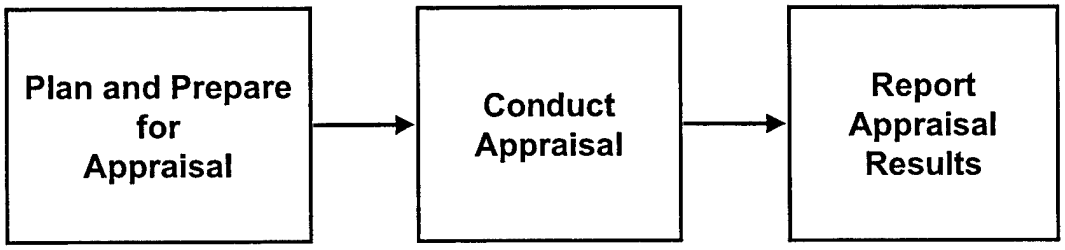


Figure 2-2. Appraisal phases.

gram office for an evaluation, contractor senior management for an assessment) to plan and prepare for the appraisal. Next, the team conducts the appraisal of each organization or significant software team member (see Part II, Figure 3-2 for definition). Depending on the appraisal method, the appraisal team may hold some form of discussion with the appraised organization. Once the appraisal is completed, the appraisal moves into the third phase, in which the team reports the identified risks to the sponsor for monitoring. As part of this third phase, the appraising team or sponsor may also provide feedback to each of the appraised organizations on their strengths, inadequacies, risks, and improvement opportunities.

For a source selection, these phases would correspond to the preparation and planning done prior to the release of the Request for Proposal (RFP); conducting the appraisal during source selection; and post-contract award activities, respectively. For appraisals done for contract performance monitoring, the phases would correspond to determining the appraisal scope and goals, conducting the appraisal, and reporting the results, such as identified risks or award fee input.

A discussion of the tasks performed during each phase is provided in the following paragraphs. Each method discussed in Subsection 2.1 has its own approach to performing an appraisal, and may slightly differ in tasks performed during a particular phase of the appraisal. For example, the questions (script) used for an SCE may not be generated until the second phase, after the team receives responses to the Maturity Questionnaire (provided to the appraised organization in the first phase), but the SDCE questionnaire will be generated in the first phase and made available to the appraised organization as part of an RFP. The reader is encouraged to review the documentation that describes each appraisal method (CBA-IPI,[1] SCE,[2] SCAMPI,[3,4] and SDCE[5]) to determine how these may differ from the discussion below. Detail on the SDCE appraisal phases is provided in Part II, Section 3 of this document.

2.2.1 Plan and Prepare for an Appraisal

Planning and preparation take place between the appraisal sponsor and the appraisal team lead, and may include Aerospace and System Program Office (SPO) internal experts, the Aerospace Software Acquisition and Process Office, SMC's Directorate for Systems Acquisition, and other appropriate internal and external Aerospace and Government personnel. These individuals jointly develop a coordinated plan for conducting the second phase of the appraisal process and may forward portions of that plan, such as a questionnaire that elicits information on various aspects of software development and software process, to the organization(s) undergoing the appraisal. The steps involved in

---

1 [Dunaway 1996] Dunaway, Donna K. and Masters, Steve, *CMM-Based Appraisal for Internal Process Improvement (CBA-IPI): Method Description*, CMU/SEI-96-TR-007/ESC-TR-96-007, April 1996.

2 [Byrnes 1996] Byrnes, Paul and Phillips, Mike, *Software Capability Evaluation Version 3.0 Method Description*, CMU/SEI-96-TR-002, April 1996.

3 [SEI-MDD 2001] Members of the Assessment Method Integrated Team, *Standard CMMI Appraisal Method for Process Improvement (SCAMPI), Version 1.1: Method Definition Document*, CMU/SEI-2001-HB-001, December 2001.

4 [Barbour 2002] Barbour, Rick, Benhoff, Melanie, Gallagher, Brian, Eslinger, Suellen, Bernard, Thomas, Ming, Lisa, Rosa, Linda, and Ryan, Charlie, *Standard CMMI Appraisal Method for Process Improvement (SCAMPI), Version 1.1: Method Implementation Guide for Government Source Selection and Contract Process Monitoring*, CMU/SEI-2002-HB-002, September 2002.

5 [AFMCPAM 1994] Department of the Air Force, HQ Air Force Materiel Command. *Acquisition Software Development Capability Evaluation, Vol. 1 and Vol. 2*. AFMC Pamphlet 63-103. HQ Air Force Materiel Command. 15 June 1994.

planning and preparing for an appraisal, either for source selection or contract performance monitoring/risk assessment, include, at a minimum, the following:

- Determine risk areas to be assessed or evaluated
- Determine resource needs
- Determine appraisal constraints
- Define the appraisal process
- Develop the appraisal plan and schedule
- Tailor the appraisal process, if needed
- Establish an appraisal team
- Inform the organizations to be appraised

2.2.2 Conduct Appraisal

Appraisals may be conducted at various sites – the appraised organization's facilities; other related organizations' facilities; the sponsor's facilities; or a location designated by the appraisal sponsor—generally at the behest of the appraisal sponsor. If the appraisal is performed at the appraised or related organization's facilities, the appraisal team is required to travel to that location to evaluate the software processes, review information and documentation, potentially hold discussions with the organization's staff, and analyze the results of their appraisal. If the appraisal is held at a sponsor's site, the appraised organization is required to supply hard and/or soft copies of information and documentation to the appraisal team, which uses this information to assess the organization's processes. In this instance, discussions may take the form of documented questions to the organization, which require documented responses and potentially more supporting documentation from the appraised organization. Information may consist of documented responses to documented questions, oral responses to oral questions, or variations on these—which also depend on the type of appraisal being performed and on the sponsoring organization's needs.

The basic steps for conducting an appraisal include the following:

- Review documentation and other information from the appraised organization
- (Optionally) Hold discussions with the appraised organization for clarification and to identify discrepancies or omissions in the provided information
- Establish findings based on the information and documentation
- Analyze findings to develop strengths, inadequacies, risks, and improvement opportunities

2.2.3 Report Appraisal Results

Appraisal results consist of the analysis of the findings from the second phase of the appraisal process. Each finding may show a minor weakness in a given area, but, when combined, indicate significant problems with the appraised organization's software processes. Results are reported to the appraisal sponsor, and secondarily to the appraised organization (if allowed by the appraisal sponsor). Steps in reporting appraisal results include:

- Provide results to the appraisal sponsor
- (Optionally) Provide feedback to appraised organization

3. Performing an Evaluation

Section 2 of this document centered on appraisals, which cover both assessments and evaluations. However, from this point forward, we will concentrate on capability evaluations, since this is the Government-focused appraisal method, and will discuss them in depth in the remainder of this document. Since the SEI has terminated support of the SW-CMM, its related evaluation method, the SCE, is not referenced within this section of the guidebook.

The rest of Section 3 will cover considerations for performing an evaluation; defining the constraints prior to selecting an evaluation method; establishing the scope, planning, and timing of the evaluation; the section concludes with a set of frequently asked questions and recommendations for performing a capability evaluation.

3.1 Considerations for Performing an Evaluation

This section provides areas for consideration in performing an evaluation: understanding the program office's need to perform an evaluation, determining the evaluation objectives, and establishing the requirements for the conduct of an evaluation.

3.1.1 Determine Evaluation Need

What is the purpose of an evaluation? The motivations for evaluating offerors during a source selection, or an organization during a contract performance period, can widely vary. For source selection, the acquiring organization is motivated by the requirements of the Federal Acquisition Regulation (FAR), and its related supplements, to choose the supplier that provides the best value for cost. When included in the best value assessment, a software evaluation helps determine which of the offerors provides the most competent software development capability. During contract performance, the acquiring organization is motivated by their business goals to determine whether their contractor is achieving program cost, schedule, and quality objectives. A software evaluation performed during the contractual period can identify where software development costs, schedule, and quality are impacting the overall program goals, and highlight areas of process improvement to aid in achieving those goals.

3.1.2 Determine Evaluation Objectives

Specific reasons for performing an evaluation during source selection differ in some aspects from the reasons for evaluating an organization during its performance period. The two methods discussed in Section 2.1 have specific objectives for an evaluation during source selection. Table 3-1 summarizes

Table 3-1. Method Objectives for Evaluation During Source Selection

SDCE	SCAMPI MIG
Provide a comprehensive description of offerors' software development capabilities	Provide discriminators between offerors regarding process capabilities
Obtain commitment to follow well-defined and planned processes	Obtain contractual commitment to use mature processes
Provide a vehicle for dialog between proposal teams and acquiring organization	Satisfy policies or regulations that apply to the acquiring organization
Reduce program risk through early focus on software capability and process	Identify risks in process capability
Emphasize importance of mature software processes to the acquiring organization	

the reasons presented in the AFMC pamphlet for the SDCE,[6] and in the SCAMPI Method Implementation Guide (MIG).[7]

When a program office has defined a statement of need for a new program, it may be in the program's best interest to perform a software capability appraisal using one of the methods described in Section 2.1. Since evaluations are time and resource intensive, the program office should work closely with the Aerospace Software Acquisition and Process Office, and SMC's Directorate for Systems Acquisition to determine the extent to which an evaluation should be done. Frequent interchanges will help both the evaluation experts and the program office determine which method, if either, best suits the source selection needs. At the same time, the evaluation team also should receive clear, documented commitment from the program office to support the evaluation's training, staffing, performance, and resource requirements.

On the other hand, not all new program starts will require a software capability evaluation. Small, precedented programs with minimal software development may not require an extensive evaluation to determine their offerors' development capabilities; these may be better served through an evaluation of contractual deliverables for software development (such as a Software Development Plan (SDP)) as part of their Contract Data Requirements List (CDRL). Again, the interaction between the evaluation experts and the program office will help determine whether an evaluation is necessary, and, if so, to what extent an evaluation will need to be performed (e.g., tailored SCAMPI or small SDCE).

SCAMPI evaluations may be applied to both new program starts, as discussed above, and contract performance monitoring (CPM). Specific objectives for CPM SCAMPIs, as listed in the SCAMPI MIG, are: to motivate the supplier to focus on contract-performance process issues, e.g., through the use of award/incentive fee; to involve the supplier team in improving process performance; to identify and manage risks in process capability; and to motivate compliance with contractual commitment to process performance. The SDCE was not developed for CPM, and has only recently been used to

6 [AFMCPAM 1994] Department of the Air Force, Headquarters Air Force Materiel Command, *Software Development Capability Evaluation*, AFMCPAM 63-103, 15 June 1994, Vol I, p. 4.
7 [SEI-MIG 2002] *Standard CMMI Appraisal Method for Process Improvement (SCAMPI), Version 1.1: Method Implementation Guide for Government Source Selection and Contract Process Monitoring*, CMU/SEI-2002-HB-002, September 2002, p. II-13.

evaluate an on-contract organization for risk identification and award fee determination. The evaluation team has not yet released the results and lessons learned from that evaluation.

3.2 Evaluation Constraints

Prior to defining the scope of an evaluation for either a new program start or CPM, the acquiring organization and evaluation team lead must define and document any constraints against the evaluation. These constraints may include cost limitations, schedule limitations, or program-specific limitations that may be applicable to the acquiring organization. The following sections discuss cost and schedule constraints, which must be addressed for any evaluation.

3.2.1 Cost Constraints

The cost estimate for performing an evaluation should include the costs to the acquiring organization for personnel effort (time charges), travel costs, training costs, and cost of resources used to perform the evaluation (e.g., equipment, software applications, secure facilities). The need to perform an evaluation, and the objectives to be achieved, must be weighed against the cost constraints of the acquiring organization. When evaluating cost constraints, the acquiring organization should consider the strategies for mitigating costs, where cost savings may be taken, and the risks involved in accepting a particular strategy. Table 3-2 provides some examples of cost mitigations, related savings, and risks.

Each of the evaluation techniques in Section 2.1 is affected by cost constraints. Personnel hours for the SDCE have been included in Part II of this document (see Section 4.3 and Section 5.4), which can be used to provide a baseline for the acquiring organization's cost estimate. The SCAMPI MIG[8] also provides some cost considerations for the acquiring organization's procurement or monitoring efforts.

Table 3-2. Cost Constraint Mitigations, Savings, and Risks

Cost Constraint	Cost Mitigation	Cost Savings	Risk
Personnel effort	Limit the scope of the evaluation	Reduce personnel effort	Potential increase to schedule Deletes pertinent questions or process areas that could highlight development or process risks
Travel expenses	Eliminate on-site evaluations	Eliminates travel costs	Precludes information gathering from developers No first-hand assessment of development and test environments
Training expenses	Use only highly experienced evaluators	Reduces training costs	Unavailability of required personnel No mentoring or training of less-experienced personnel for other evaluations
Resource costs (e.g., facilities, computer resources)	Limit or share resources for the evaluation	Reduces resource costs	More difficult to implement source selection access restrictions Schedule delays due to unavailability of resources

8 [SEI-MIG 2002], Subsection 1.1.2.

3.2.2 Schedule Constraints

The major schedule constraint confronting acquiring organizations is the use of software evaluations in a streamlined source selection. Streamlined acquisitions have extreme limits to their source selection schedules, usually 90 to 120 days from receipt of proposals to contract award. For evaluations to be done in this limited time, the acquiring organization and evaluation team lead must determine which method, and application of that method, will be sufficient for meeting their needs and objectives.

In order for an evaluation to be performed adequately in a constrained source selection schedule, there are two primary options available for the acquiring organization to consider: tailor the evaluation method and process such that it fits within the source selection time frame; or perform the evaluation prior to the start of source selection, during the earlier acquisition period. Both of these methods have benefits and shortcomings, and both allow for the evaluation results to become part of the acquiring organization's selection criteria. Tailoring reduces the amount of information to be reviewed as part of the source selection, thus reducing the time required for a team to complete the review, but also may eliminate access to information that would highlight development and process risks. Performing the evaluation prior to the source selection potentially allows for additional information gathering and time to review, but also requires additional contractual vehicles for the offerors to provide information during the current acquisition period. As part of the following source selection, the evaluation team will still need to review pertinent documents from each of the offerors [e.g., updated SDPs, updated Integrated Master Plan (IMP)] and provide feedback to the source selection team on the quality and maturity of the software processes.

Part II of this document provides several techniques for performing an SDCE under streamlined acquisitions timelines. Similar applications of the SCAMPI, with the same contractual constraints, are possible. The acquiring organization and evaluation team lead should discuss both options and determine which method is best suited to the program.

Although CPM evaluations are not as limited as acquisition evaluations, the program office must make some consideration of schedule, both for themselves and their contractor organization. The program office must evaluate their internal and external resource loading prior to scheduling a CPM evaluation to eliminate task overlap and remain within their budget and schedule. They must also determine the impact to the contractor's on-going development prior to scheduling an evaluation. Since the contractor team members must be pulled from scheduled daily tasks to respond to evaluation questions or track down supporting documentation, they may not be able to complete their normal work obligations and thus fail to meet internal or external contractual milestones. Work being performed by associate contractors and subcontractors should also be factored into the evaluation schedule, and any potential impact noted. It is incumbent upon the acquiring organization and evaluation team lead to ensure that this type of evaluation is coordinated with the both the acquirer's and contractor's upper management to limit its incursion into program's work objectives and schedule.

3.3 Evaluation Scope, Planning, and Timing

Establishment of the evaluation scope, the evaluation plan, and the timing of the evaluation are all closely linked to the evaluation considerations and the constraints. The cost and schedule constraints

may limit the scope of the evaluation, the schedule constraints may restrict the timing of the evaluation, and the planning for the evaluation must include all the relative decisions made by the acquiring organization or program office, the evaluation team lead, and possibly the contractor team/offerors. Since an evaluation is driven by the needs of the acquiring organization/program office team, they must reconcile the interaction of all of the needs, requirements, and constraints in order to derive the scope, plan, and timing of their evaluation.

3.3.1 Determine Evaluation Scope

The evaluation scope consists of the scope of the model being used and the scope of the information to be gathered to support the findings of that model. Generally, the scope of the model is driven by the needs and risks of the acquiring organization, and the scope of information is driven by their constraints. Essentially, evaluation scope condenses to "What pieces of the selected model cover our needs and risks?" and "How much time, money, and resources do we have to perform the evaluation?"

For example, the SCAMPI model may be tailored to examine specific process areas (PAs) that align with the acquiring organization's identified needs and risks. The scope of information that is available to the SCAMPI team to evaluate the contractors' or offerors' processes with respect to the PA goals may be limited by cost constraints (e.g., no travel funding for on-site interviews, limited personnel effort dollars), schedule constraints (e.g., the process evidence is limited to one instance to reduce the time required by team members to perform the review) or contractual constraints (e.g., information is available only for the prior phase of the life cycle, or subcontractors may not be evaluated due to proprietary development restrictions). Similar scoping for the SDCE is discussed in Part II, Subsection 3.1.

3.3.2 Develop Evaluation Plan

Development of the plan for an evaluation requires time and resources. The evaluation team lead (and possibly a small staff) may need to allocate several weeks of effort for planning, depending upon the type of evaluation being performed (new program start vs. CPM), the complexities involved (e.g., teaming arrangements among the contractors), and the amount of tailoring for the scope and instructions. However, the evaluation plan development and its implementation may be impacted by the evaluation cost constraints (such as limited funds) and schedules (such as deadlines and milestones for evaluation completion), which the team lead must take into account as discussions with the acquiring organization and plan development proceed.

The plan's contents are dependent upon the method employed for the evaluation and type of evaluation being performed. First, the plan should specify the model being used for the evaluation (SDCE or SCAMPI), how it will be tailored, how the instructions for the model will be generated and approved, and when it should be released to the contractor or offerors. For a new start program, the plan must include a schedule based on the source selection acquisition strategy, and assign personnel based on the schedule limitations. For a CPM evaluation using the SCAMPI, the plan schedule must work within the overall program schedule to preclude development delays. The overall schedule for completing the evaluation must include startup tasks, the formal evaluation process, and final report generation. Other sections of the plan should address, at a minimum, the staffing requirements for the

evaluation; the resources required for the evaluation; the site visit process (as needed); the process for handling proprietary contractor or offeror documentation; the agreements made with the acquiring organization, along with any constraints on the evaluation; and the decision as to whether to evaluate each significant software team member (e.g., company, division) or evaluate the team as a whole. Finally, all acquirer and support stakeholders in the evaluation should provide documented concurrence with the plan.

3.3.3 Evaluation Timing

An evaluation can be performed at diverse points in time during a program's life cycle, or it may be limited to specific points in the life cycle. For a program that is in its development phase, a CPM evaluation may be done at specific milestones to support award fee assessments, or to determine that the contractor team is following their contractually required processes. At any time, when the program office notes a significant number of errors in the contractor's products, a CPM evaluation may be used to find root causes of the errors, uncover previously unidentified risks, and highlight areas for process improvement. For a program that is initiating a new start contract, the evaluation may be performed in support of the source selection, but may be done prior to or outside of the source selection itself.

Award fee milestone evaluations are performed using the SCAMPI method, and can provide benchmarks through the program life cycle to show areas of successful process improvement. These types of evaluations are scheduled in the program as part of contract award, and agreed to by the contractor, such that minimal disruption of the program development occurs. The milestone evaluations can also be used to determine whether or not the contractor has successfully applied their contractually required processes. Award fee may be increased or withheld depending upon the benchmark improvements and process implementation.

The program office may require unscheduled CPM evaluations when a contractor incurs multiple risks or failures due to poor process implementation. These evaluations may be scheduled on short notice, and may use an independent team of evaluation experts with little program knowledge. Their findings are reported to the sponsoring program and to the evaluated contractor team for process improvement actions and risk monitoring. Again, these types of evaluations are performed using a SCAMPI method due to its applicability to contract performance monitoring.

For source selection, two applications of an evaluation must be considered: the evaluation as part of source selection, or the evaluation prior to source selection. If the evaluation can take place outside a source selection, a more thorough (deeper or broader) evaluation is possible. Within the source selection timeframe, the content of the evaluation may be constrained by schedule and resources.

An evaluation may be performed before a source selection if the acquisition strategy has adequately addressed its use. The use of the SDCE outside of source selection is addressed in Part II of this document (see Section 3.1). The SCAMPI method, with its larger support structure, can be implemented outside of source selection by either a program-office-led evaluation team, or by an independent team of evaluators with little knowledge of the program or the components of the new program start. Furthermore, the independent team might not be part of (read into) the source selection, and

therefore might have no access to the offerors' proposals or related program information and no insight into the processes proposed for the project under bid. In the independent evaluation, the findings are provided to the source selection team to include in the final proposal reports and briefings. Additionally, any organization that is planning to bid on the program, as a prime contractor or significant software team member, may be required to undergo a SCAMPI evaluation during the time period between the Commerce Business Daily (CBD) announcement and the start of source selection.

The timing for SCAMPI evaluation preparation for source selection differs slightly from that presented in Part II for the SDCE, specifically in the release of the Maturity Questionnaire and in the documentation review and interview process. The Maturity Questionnaire is provided to potential offerors prior to or concurrent with the release of the Request for Proposal. Responses are returned to the acquiring organization for review and potential downscoping of the interview script after the start of the source selection. Documentation review and the interview script are used to gather information on the offerors' use of processes. Figure 3-1 depicts the SCAMPI method's inclusion in the source selection process.

3.4 Frequently Asked Questions Regarding Appraisals

Questions on the appraisal methods, certification of appraisals, and reasons for appraisals have commonly been asked of the Aerospace acquisition organization. We document these questions and responses here for the reader's information.

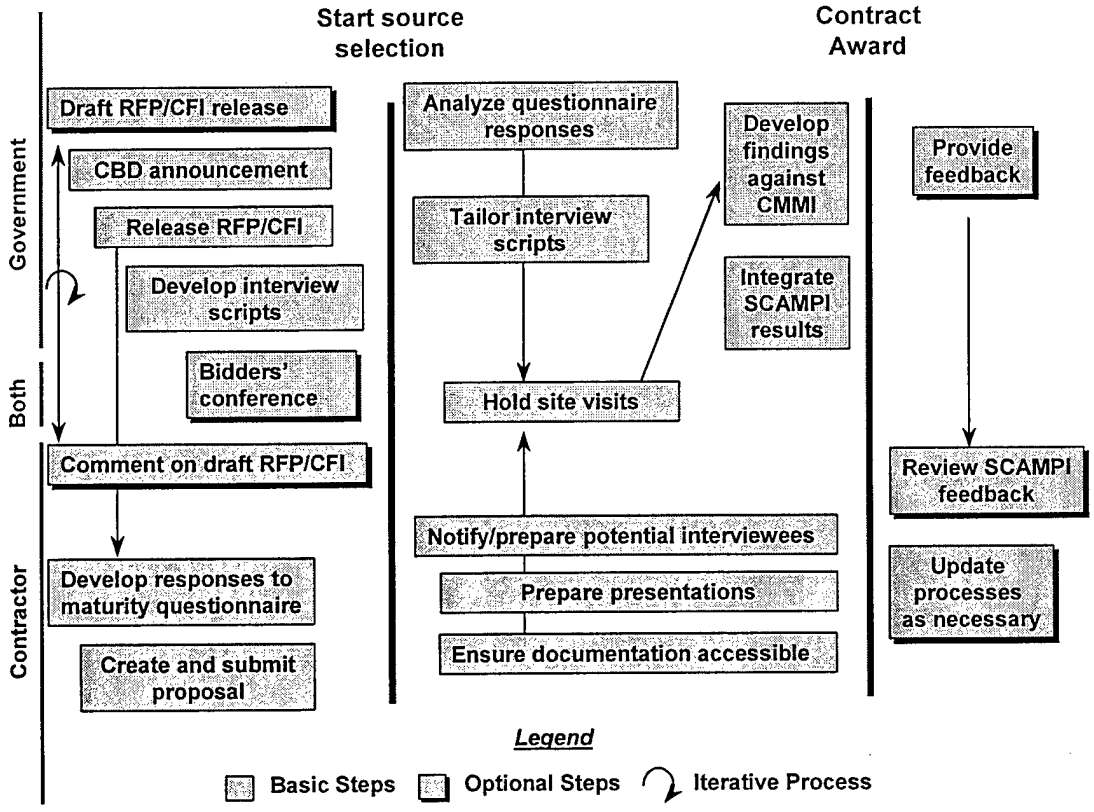


Figure 3-1. The use of SCAMPI in the source selection process.

Q. I'm doing a new program start at SMC with some unprecedented features and it's pretty software intensive. What type of evaluation should I do?

A. You should probably go forward with a targeted SDCE—one that is tailored to align with your program's risks and needs. Since your source selection may be time restricted (streamlined), a large SDCE or full SCAMPI would not be possible. You could also do a tailored SCAMPI evaluation, but scheduling the limited staff within Aerospace may conflict with your source selection schedule.

Q. I've just received the proposals for my new program start and each of the offerors claims to be at CMMI Level 4. Is this type of information useful to me for my source selection evaluation?

A. Generally, internally derived assessment levels indicate a higher level of process maturity than the organization actually has. When a contractor performs a self-assessment, they tend to give themselves more credit for process maturity than would be given by an independent, external evaluation team. Furthermore, only a very limited number of (usually cherry-picked) projects is examined in an assessment. SEI representatives have stated on numerous occasions, such as at the Software Technology Conferences, that many self-assessments place the contractor one or two levels higher than an evaluation would.

Q. I've just received the proposals for my new program start and each of the offerors claims to be at CMMI Level 4, with Government participation on the assessment team. Is this type of information useful to me for my source selection evaluation?

A. The inclusion of a Government representative on the assessment team probably doesn't add much to the rating. A minimum of two Government representatives is preferred because they are then less likely to be pressured into consensus by the contractor team. The Government representatives are added to the team to give the results more credence when reported in a document (like a proposal). However, the value added of the Government representative depends wholly on what that person contributed as part of the team. Since the rating does not provide any dissenting opinions, or give any indication of the expertise of the Government representative, you have very little to go on. You may want to ask for the Government representative's name and phone number, and to contact him/her independently for confirmation of the results. You would also want to ask the contractor for the Appraisal Disclosure Statement (ADS), the complete final findings presentation, and the non-attributable worksheets.

Q. Should SMC RFPs require a development contractor to be "certified" or "SEI certified" to at least Level 3?

A. "Certification" of a SW-CMM or CMMI Level is impossible. No Level "certificate" or piece of paper exists. RFPs therefore cannot contain "certification" language.

Q. Can a contractor's "certification" be examined?

A. Unlike ISO certification, no certificate is given to an organization. No organization exists that is authorized to issue a SW-CMM or CMMI Level certification.

Q. Can a contractor's SW-CMM or CMMI Level be checked with the SEI?

A. The SEI does not certify any development organization. SEI maintains a database of organizations and their reported SW-CMM levels. SEI guarantees anonymity to organizations that report their levels. SEI's database cannot be accessed. SEI uses this database to report on

process improvement across industries. A list of publicly reported levels with dates of reports is at *http://www.sei.cmu.edu/sema/published.ml.html*, with disclaimers, as shown below:

> "The SEI is continually being asked for names of organizations at various maturity levels. As part of our non-disclosure agreement with these organizations, we cannot provide the names of the organizations and their maturity levels based on the data stored within the Process Appraisal Information System (PAIS). ..."
>
> "However, we have compiled a list of organizations that have publicly announced their maturity level. This information has been gathered from publicly distributed articles reporting an organization's maturity level. . ."
>
> "Disclaimer
>
> Please be aware of the following issues regarding this list.
>
> 1. The information in this list is from publicly available sources.
> 2. The Software Engineering Institute (SEI) does not certify organizations at maturity levels.
> 3. The SEI does not confirm the accuracy of the maturity levels reported in the noted sources.
> 4. This list of Published Maturity Levels is by no means exhaustive.
> 5. The SEI does not use information stored within its Process Appraisal Information System to produce this list.
> 6. The SEI does not use information obtained in confidence to produce this list."

Q. How does a SW-CMM Level relate to other software process information provided by the contractor?

A. There is some overlap between the CMM models and ISO 9000. What ISO 9000 requires is generally easier to achieve than the requirements for CMM and CMMI models. Examining the corporate and project policies, procedures, plans, and project artifacts with respect to each other and with respect to the model requirements provides information about how closely related they are.

Q. What happens when there is a team of contractors?

A. The whole team developing software needs to be evaluated for this project. Each significant software team member's processes need to be examined to determine strengths, weaknesses, risks, and improvement opportunities. The way each team member organization's processes interact with the other organization's processes should be evaluated.

Q. What happens to their processes when one company acquires another?

A. Some organizations go through turmoil while they sort out which processes to use. Generally, some parts of the new organization retain their processes, and others adopt processes from other parts of the organization. Even though the processes may be improved processes, it will take the new organization time to adapt.

Q. How long before an assessment level expires?

A. There is not a period like with ISO 9000 with reviews every six months for three years. The maximum period for reusing an evaluation should be no longer than two years. Organiza-

tions sometimes backslide when they have finally made their goal level. Senior management is required to continually support and check on the processes. In the rush to stay on schedule, some managers forget why the processes are in place and cut corners. The immediate savings toward a short-term goal often makes the project late with more defects. An organization can slip a level in less than a year. Organizations that reached Level 4 and 5 because their senior management learned the hard way the reasons for the processes are less likely to backslide.

Q. How good are a contractor's processes if they are a level 3 (or 4 or 5)?

A. The processes are only as good as the ones used on the projects appraised. Most likely, the projects with the best processes were selected, and there is no guarantee that those processes would be used on your project.

Part II. USING THE SDCE AS AN APPRAISAL TECHNIQUE

1. Background on the SDCE

The Software Development Capability Evaluation (SDCE) is a structured methodology used by Air Force Space and Missile Systems Center and National Reconnaissance Office (SMC/NRO) organizations during source selection to gauge a contractor's ability to develop mission-critical software.[9] The SDCE was developed by an Air Force-sponsored Process Action Team in response to a perceived need for a tailorable, program-focused, software evaluation method to assess software-related areas of that program, specifically during source selections. As such, the SDCE focuses on each bidding team's planned software development efforts for a specific program's Request for Proposal (RFP), and is used to determine whether a bidding team has proposed the use of demonstrably mature software development processes.

The SDCE consists of a model and a process for conducting the evaluation against the model. The SDCE model, composed of questions and related criteria, provides a standardized approach for assessing the maturity of each software development contractor for a specific program. The SDCE process, consisting of a series of evaluation activities, provides a standardized technique for determining the strengths, inadequacies, and risks associated with each contractor's proposed software development processes. A complete description of the original SDCE model and process can be found in the reference in footnote 10.

Recent Air Force and DoD changes have had an impact on the SDCE model and process, and in response to these changes in policy and regulations, the SDCE has undergone modification. In June 1999, the Air Force Federal Acquisition Regulation Supplement (AFFARS) was revised to simplify the source selection process, which included the revision of technical areas and factors into specific factors and subfactors, respectively. In October 1999, the Undersecretary of Defense for Acquisition, Technology and Logistics amended the DoD policy for Acquisition Category (ACAT) I and ACAT IA contracts. This policy stipulated that offerors on these contract types must be evaluated at Capability Maturity Model (CMM) for Software (SW-CMM) Level 3, or its equivalent level using a DoD-approved tool. In support of the policy change, the Deputy Undersecretary of Defense for Science and Technology [DUSD(S&T)] called for a team of industry, military, government, and Federally Funded Research and Development Center (FFRDC) personnel to assess the SW-CMM Level 3 methodology and its relationship to other evaluation techniques, i.e., the SDCE. The outcome of the

9 While the SDCE is also used at Air Force Aeronautical Systems Center, this guidebook focuses on its use in the SMC/NRO (National Reconnaissance Office) environment.

10 [AFMCPAM 1994] Department of the Air Force, Headquarters Air Force Materiel Command, *Software Development Capability Evaluation*, AFMCPAM 63-103, 1994, Vol I, pp. 6-8.

assessment was a DoD-approved SW-CMM Level 3-equivalent SDCE "core" questionnaire, consisting of 130 questions and related criteria that may be applied on ACAT I and IA programs undergoing source selection. In addition, the team developed a set of general requirements for process evaluation methods and their applications.

Modifications to the SDCE are discussed at a high level in the sections below, but are covered in greater detail in Section 5 of this document.

2. Software Development Capability Evaluation Model Structure

The SDCE model structure is a hierarchy of functional areas, critical capability areas, critical capabilities, and questions and criteria associated with program management and systems and software engineering processes. The questions and criteria relate to capabilities in program management, systems engineering, and software engineering that have been identified by program offices and the engineering community as risk areas during software development. Figure 2-1 illustrates the hierarchy of the SDCE structure using the Peer Review Planning Critical Capability as an example.

2.1 The Functional Areas

The SDCE model comprises six functional areas: Program Management, Systems Engineering, Software Engineering, Quality Management and Product Control, Organizational Resources and Program Support, and Program-Specific Technologies. These topics remain unchanged by the updates due to policy implementation. Figure 2-2 provides a top-level view of the model and highlights the Critical Capability Areas (CCAs). The paired numbers provide the number of questions associated with a particular SDCE component (bottom or right-hand number) and the number of questions in that component that are "core" questions (upper or left-hand number) in the SW-CMM Level 3-equivalent SDCE.

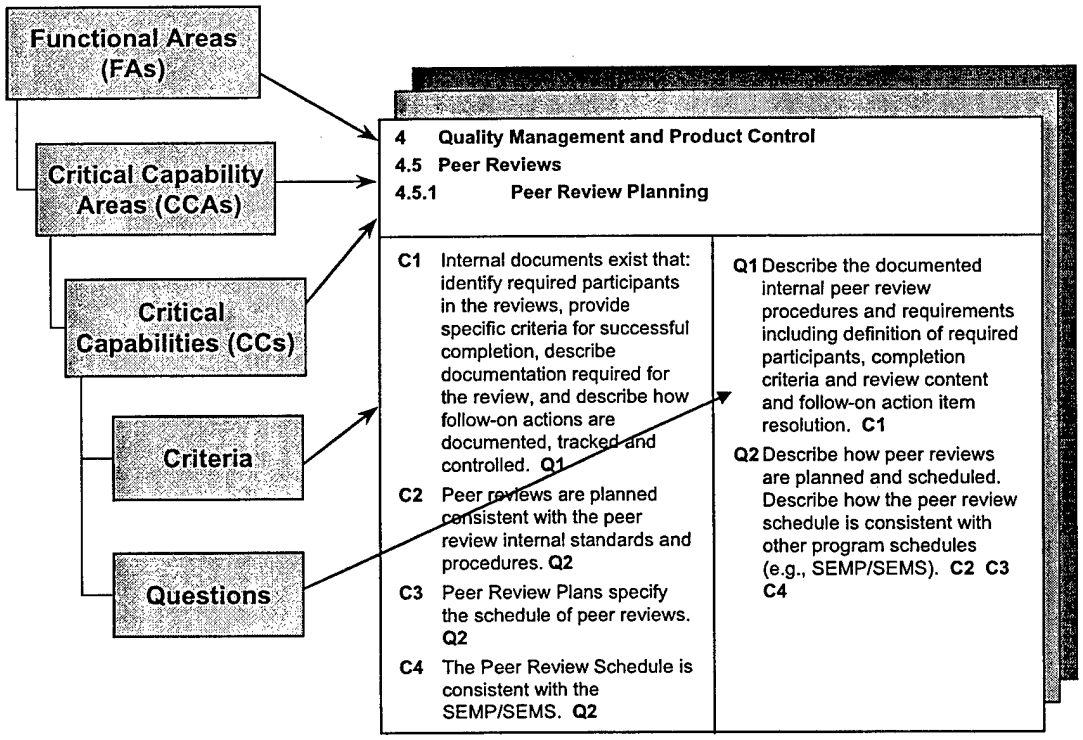


Figure 2-1. Example of the SDCE model structure hierarchy.

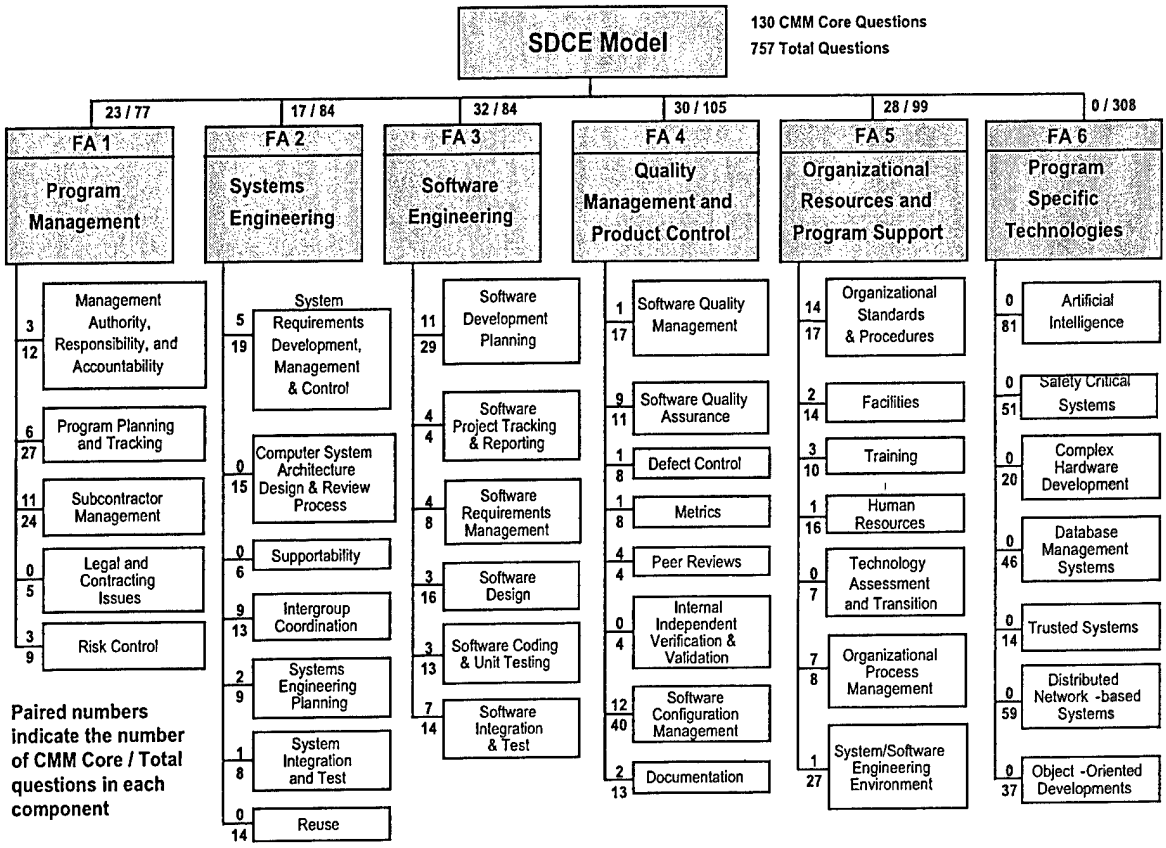


Figure 2-2. SDCE model—top level.

Each Functional Area focuses on aspects of the software engineering discipline. For example, Functional Area 1 evaluates a contractor's capability to manage the software development effort as part of its overall program management. Functional Area 2 provides insight into the integration of software engineering with the program's systems engineering approach. The Software Engineering Functional Area 3 looks for mature software development techniques over the software development life cycle. Functional Areas 4 and 5 cross the three previous FAs to evaluate quality, configuration management, and organizational support. Functional Area 6 focuses on new technologies that may require complex engineering support. However, the software engineering for these new technologies is covered by FAs 1 through 5 as well. FA 6 may require the SDCE team to have personnel with specific expertise in the new technologies in order to perform the evaluation.

2.2 The Critical Capability Areas (CCAs)

The SDCE CCAs are shown as the unshaded boxes under the Functional Areas in Figure 2-2. Each of the CCAs has a specific focus regarding its relationship to software engineering and the software development life cycle. Section 3 of Air Force Material Command Pamphlet (AFMCPAM) 63-103 [AFMCPAM 1994] provides a description of each of the CCAs and their individual areas of evaluation.

Although no new CCAs were added as part of the SW-CMM Level 3 equivalence development, several were substantially changed by adding new questions and criteria or by modifying existing questions and criteria. These CCAs are:

1.1 Management Authority, Responsibility, and Accountability

1.2 Program Planning and Tracking

1.3 Subcontractor Management

2.5 Systems Engineering Planning

3.1 Software Development Planning

4.7 Software Configuration Management (SCM)

5.1 Organizational Standards and Procedures

5.3 Training

5.6 Organizational Process Management

2.3 The Critical Capabilities, Questions and Criteria

Figures 2-3 through 2-10 show the CCAs and CCs for the six Functional Areas depicted in Figure 2-2. In each figure, the numbers along the horizontal lines provide the number of questions associated with a particular CC (bottom number) and the number of questions in that CC that are "core" questions (upper number) in the SW-CMM Level 3-equivalent SDCE.

For SDCEs not required to use the core questions and criteria, a set of frequently used questions has been developed, which includes tailoring to provide a qualitative assessment of the offeror's processes and to delete obsolete references. These "basic" questions are included in Appendix A.

As part of the update for the Level 3-equivalent SDCE, a new Critical Capability was added to evaluate the institutionalization of software process within the bidding organization. This new CC is labeled CC 5.1.4, Process Institutionalization. The SW-CMM Level 3-equivalent core set of questions and criteria, including the new CC and any other new or modified questions and criteria can be found in Appendix C. The SW-CMM Level 3-equivalent core set of questions and criteria was intended for use on ACAT I or IA programs, or if the program desires an evaluation of Level 3 equivalence. Level 3 equivalence is no longer required for ACAT I and ACAT IA DoD programs or for any space program per recent policy changes.

The original SDCE questionnaire (questions and criteria) can be found in AFMCPAM 63-103, Volume 1, Section 5. New and modified questions for the SDCE, which have been used on AF program evaluations, are included in [Haddad 1998a].

Aerospace has also developed a set of questions and criteria that pertain specifically to the use of commercial off-the-shelf (COTS) software components in software development; these questions and criteria are included in Appendix D of this document.

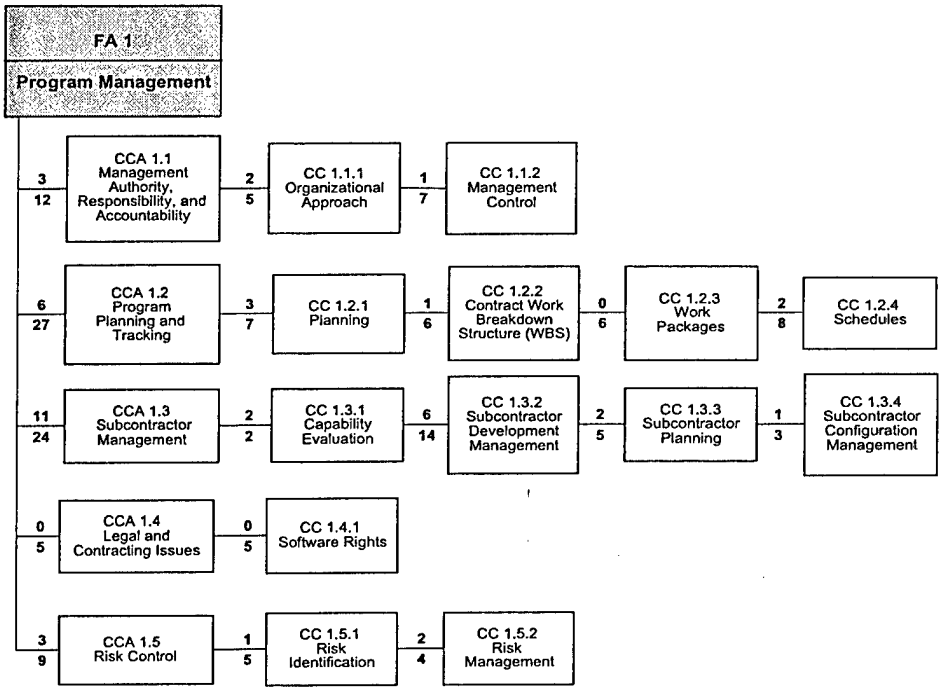


Figure 2-3. Critical capabilities for Functional Area 1.

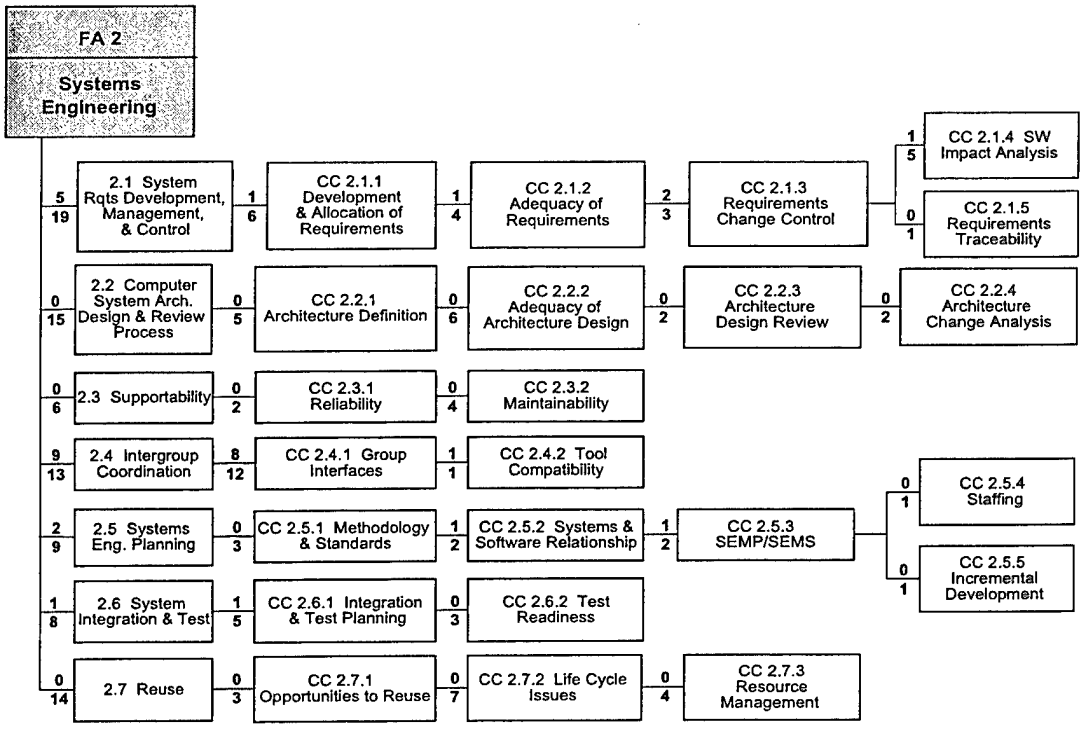


Figure 2-4. Critical capabilities for Functional Area 2.

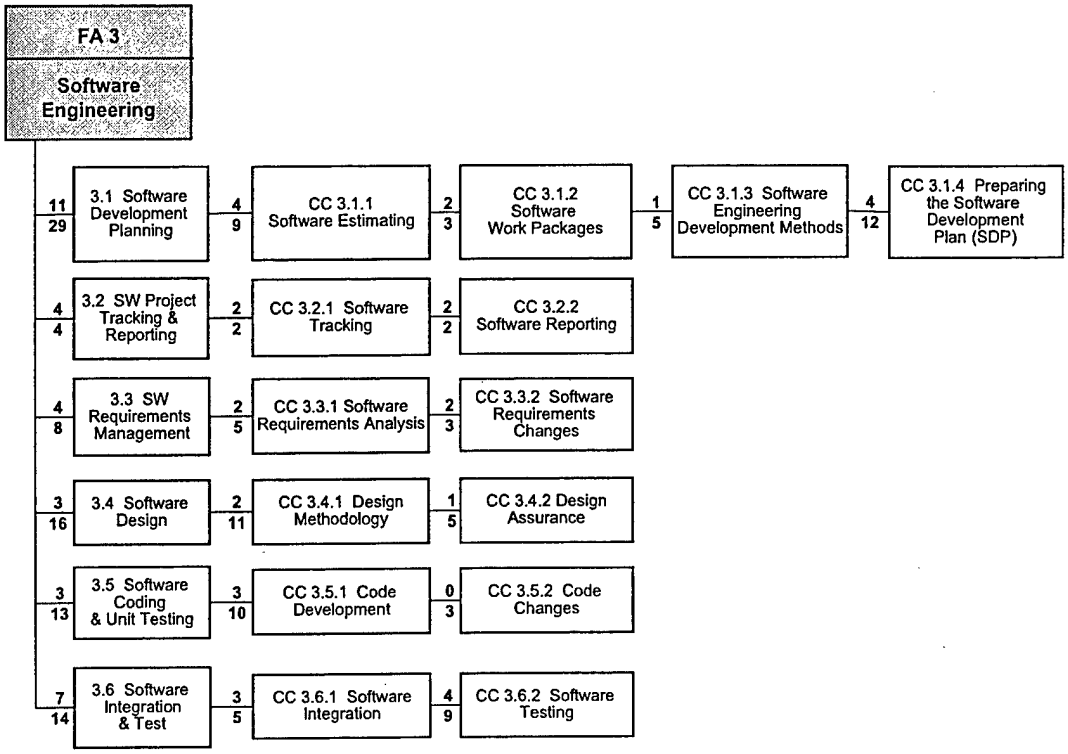


Figure 2-5. Critical capabilities for Functional Area 3.

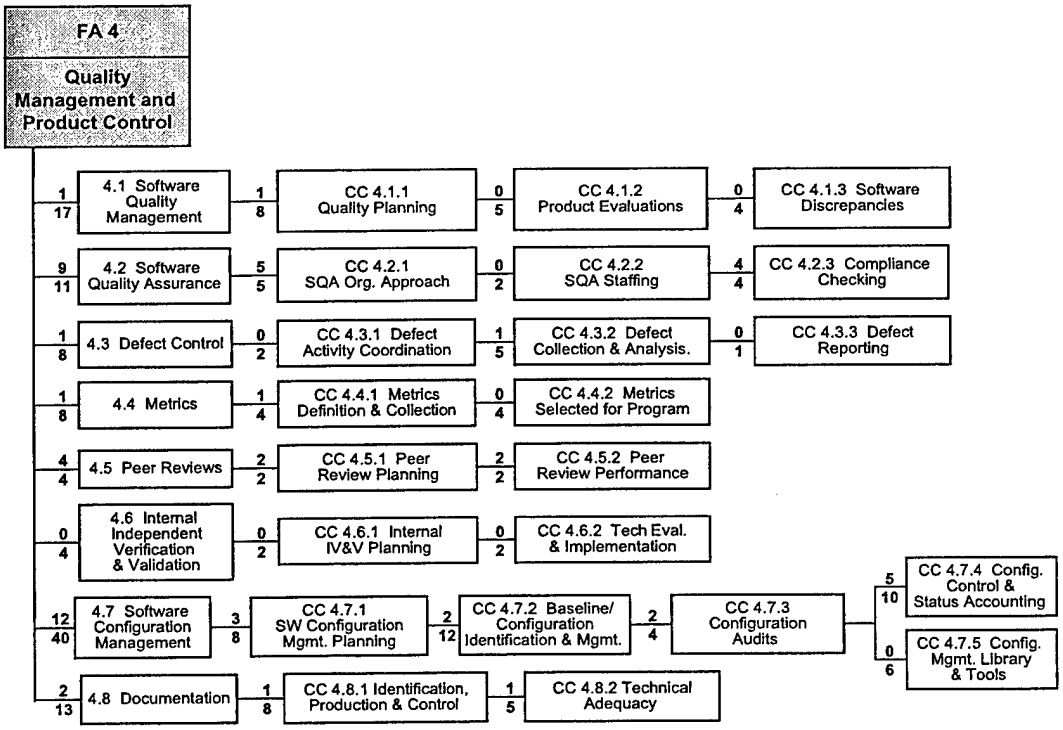


Figure 2-6. Critical capabilities for Functional Area 4.

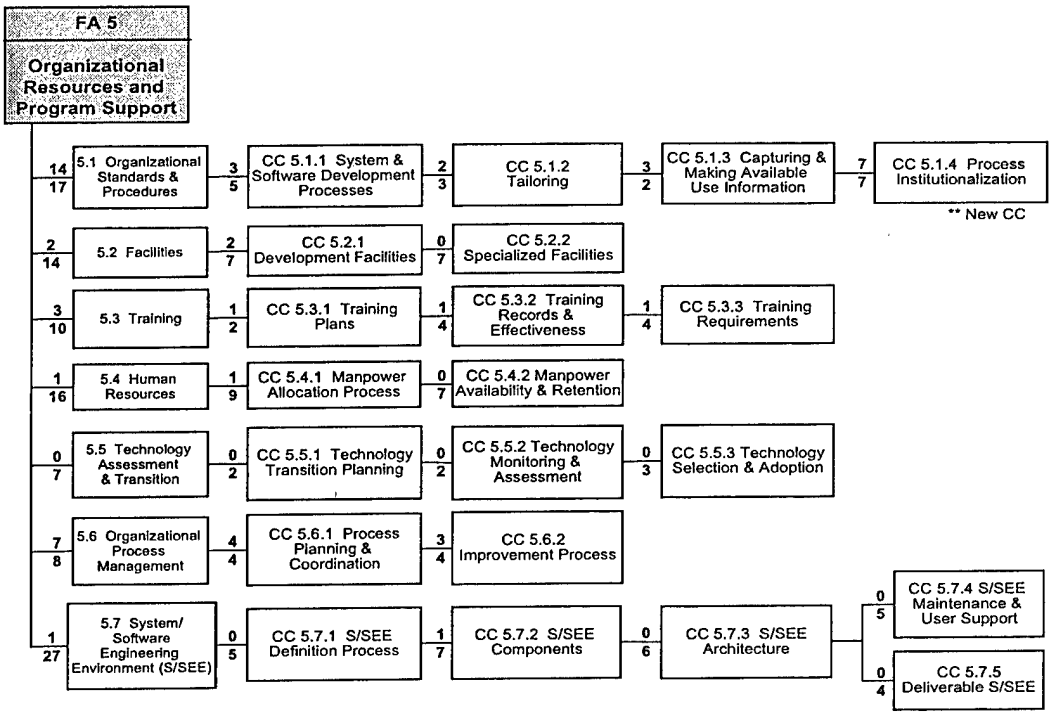


Figure 2-7. Critical capabilities for Functional Area 5.

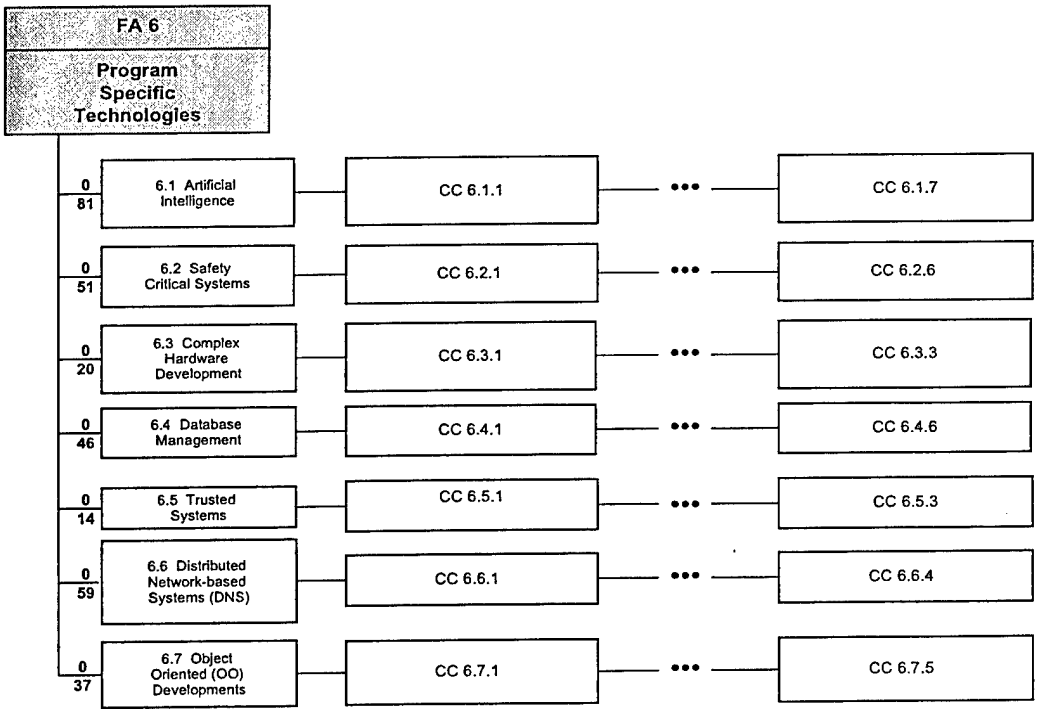


Figure 2-8. Critical capabilities for Functional Area 6 (top level).

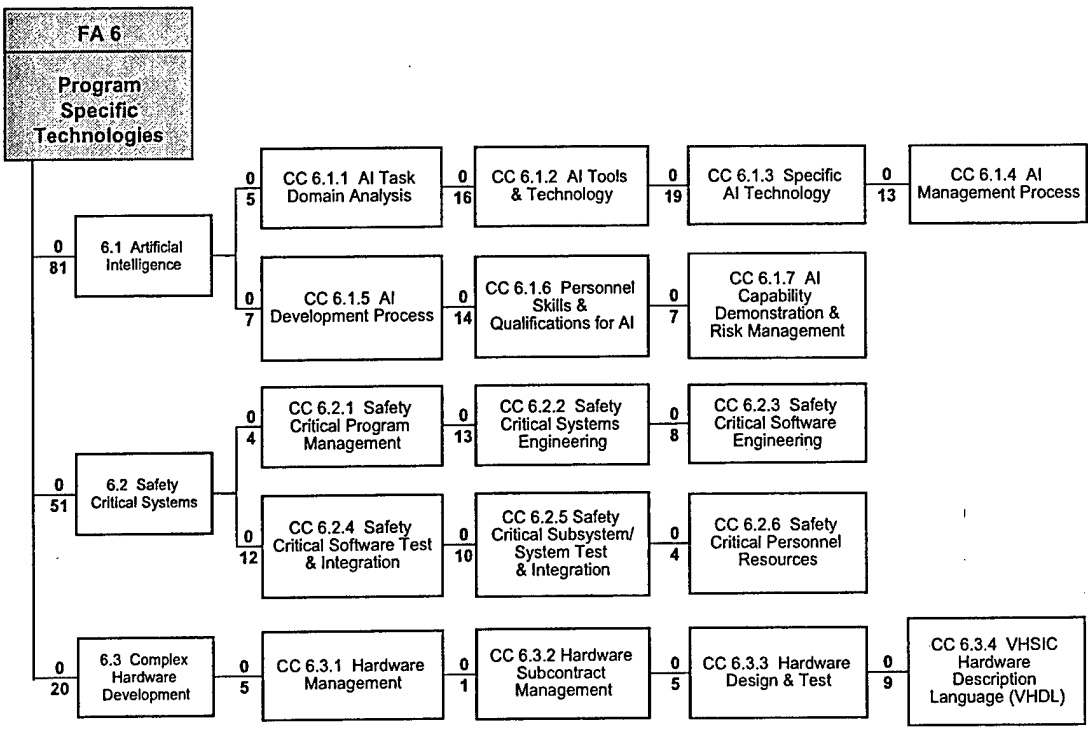


Figure 2-9. Critical capabilities for Functional Area 6 (Part I).

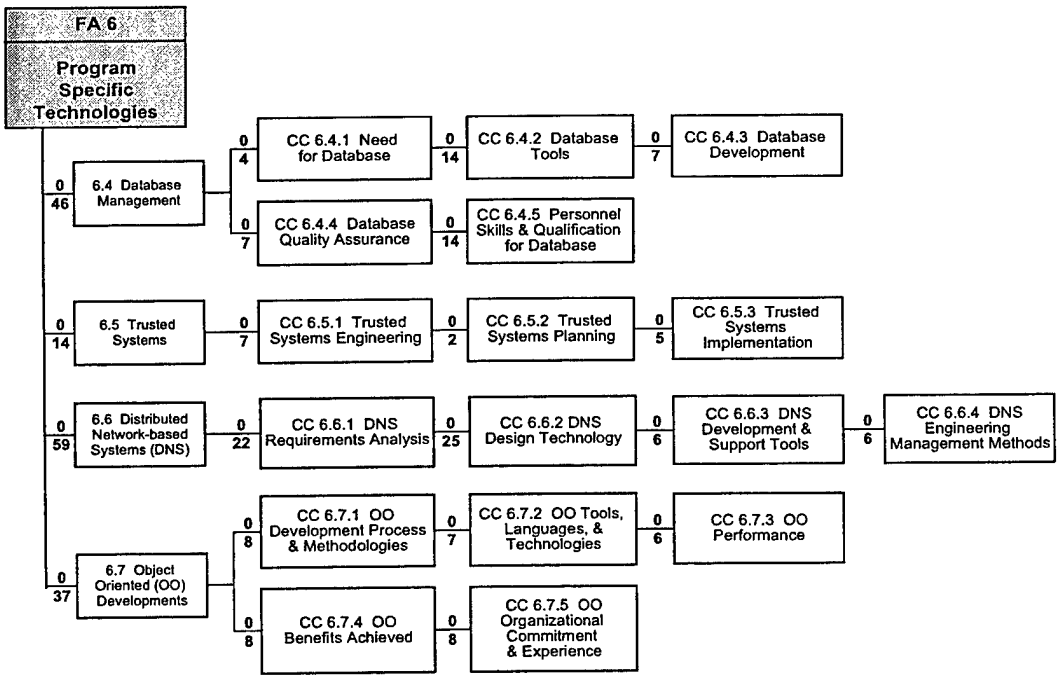


Figure 2-10. Critical capabilities for Functional Area 6 (Part II).

3. Software Development Capability Evaluation Process

The process followed by the SDCE team is depicted in Figure 3-1. The three phases in the figure nominally relate to the three phases of the source selection process: pre-RFP release, the source selection proposal evaluation, and post-contract award. Prior to the release of the RFP, the SDCE team provides assistance to the program office to plan and prepare for the SDCE to take place during proposal evaluation. During the source selection, the SDCE team conducts the evaluation of each offeror's response to the questionnaire, optionally holds some form of discussion with the each offeror, and integrates the team's results with the rest of the source selection evaluation. Following contract award, the SDCE team can provide feedback to both the successful and unsuccessful offerors and transition the risks identified during the evaluation to the program office for monitoring. Each of these phases is discussed in the sections below.

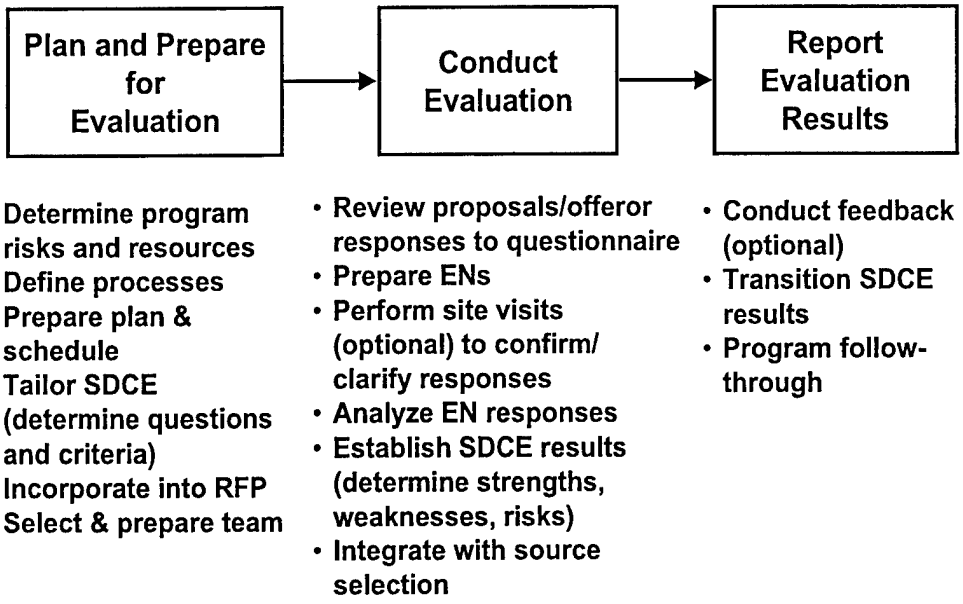


Figure 3-1. SDCE team process.

3.1 Plan and Prepare for an SDCE

3.1.1 Determine Program Risks and Resources

When a program is considering a new procurement related to a defined need, the program office performs an assessment of the high-level risks associated with the acquisition of a system or systems to meet that need. When considering a software-intensive system acquisition, the program office should strongly focus on the software portion of that system, particularly since software development is considered an area of high risk on software-intensive system acquisitions.[11] Coordination of the effort to determine the software system risks must include the expertise of the SDCE Focal Point to ensure that: (1) the benefits of performing the SDCE as a risk mitigation technique are understood by the program office acquisition team; (2) the pitfalls of not performing an SDCE are made known to the program office; and (3) the software risks are identified prior to development of the SDCE questionnaire and instruction set. It is, therefore, incumbent upon the program office to ensure that the SDCE Focal Point is contacted and available for the planning and preparation of the RFP and SDCE. Alternatively, should the SDCE Focal Point become aware of a new procurement that may require an evaluation, it is incumbent upon the SDCE Focal Point to enlighten the program office on the applicability and benefits of performing an SDCE. Finally, the program office, SDCE Focal Point, and the Focal Point's designated SDCE team leader must coordinate to provide sufficient resources to complete the SDCE planning, training, and formal evaluation.

3.1.2 Define Processes

Based upon the anticipated number of bidders, funding, source selection schedule, and personnel, the SDCE team leader establishes the SDCE process as part of the overall source selection process. The process may provide for early notification to the offerors that an SDCE will be required as part of the source selection, i.e., releasing Section L and Section M paragraphs for the SDCE as part of a draft RFP with placeholders for the SDCE questionnaire. The team leader also reviews the acquisition strategy and schedule to determine whether site visits can be part of the evaluation process. The SDCE process will also include information on the source selection process, such as categories for criteria assessments (strengths, inadequacies, risks), taking into consideration the acquisition rules regarding discussions with the offerors, release of evaluation notices (ENs), and assessment of EN responses. The process for completing the evaluation is formed, which determines how each question and criterion will be assessed, e.g., by team consensus on the response to each criterion, "two pairs of eyes" reviewing each response and coordinating their findings, or by individual team member findings. Finally, the team leader establishes the training process for the SDCE evaluation team and selected source selection team members, e.g., Factor and Subfactor chiefs.

3.1.3 Prepare Plan and Schedule

As the process solidifies, the SDCE team leader develops the plan for performing the SDCE, including team selection and preparation, and creates an SDCE evaluation schedule based on the source selection need dates. Part of the planning includes the placement of the SDCE in the source selection structure. In general, this will be as a subfactor or element under the Mission Capability factor for

---

11 [STSC 2000] Department of the Air Force Software Technology Support Center, *Guidelines for Successful Acquisition and Management of Software-Intensive Systems—Version 3.0*; May 2000. Volume 1, Part I, Chapter 2.

source selections performed under Air Force FAR Supplement 5315, Contracting by Negotiation, Subsection 5315.304(c).[12] As a subfactor, the SDCE results will have a greater influence in the outcome of the source selection; as an element under a subfactor, the SDCE results are combined with evaluation results of other elements, with an overall weakening of the SDCE's influence. Thus, the SDCE team leader must strive to convince the program office either to leave the SDCE at a higher subfactor level, or to minimize the number of overshadowing elements that are also under the SDCE's subfactor. For the SDCE evaluation schedule, the start date is confirmed with the program office acquisition leader, dates are set for site visits (if performed), and the overall time limit for completing the SDCE is established.

3.1.4 Tailor SDCE (determine questions and criteria)

The SDCE evaluation schedule directly impacts the size of the SDCE questionnaire and the size of the evaluation team. A short source selection schedule precludes a large number of SDCE questions and criteria unless a large team is available to complete the work within the stated time. This can be extremely costly to the program office, especially if site visits are performed. Hence, the SDCE questionnaire (questions and criteria) is tailored by the SDCE team leader and software-cognizant program office personnel to provide a "best-fit" to the software risk areas identified earlier in the preparation process, while meeting the source selection timeline and personnel resource constraints. A recommended approach to developing a tailored SDCE questionnaire is to start with a small set of questions and criteria (see Appendix A), then expand the set, using the SDCE pamphlet and other sources of questions and criteria[13] to cover any other identified software development risks.

3.1.5 Incorporate into RFP

When an SDCE is to be performed as part of a source selection, the instructions to the offerors must be very specific as to what additional information is to be provided with the proposals. The instructions in RFP Section L that require the offerors to respond to the SDCE will also contain the reference to the SDCE instructions and questionnaire as part of the RFP. Figure 3-2 contains sample language to be used in Section L of the RFP for the SDCE. Figure 3-3 provides a sample of the evaluation criteria to be used to assess SDCE results in relation to other parts of the proposal.

As stated earlier in this document, the structure for RFP Sections L and M is fixed at four factors (Mission Capability, Cost/Price, Past Performance, and Proposal Risk). Figure 3-4 depicts an L and M structure and the desired location of the SDCE in the Section L instructions and Section M criteria. If possible, the SDCE should be under the Mission Capability factor as a subfactor to enhance its weight in the overall source selection process. The program office and SDCE team leader may have the situation where the SDCE is allocated to an element under a technology subfactor, e.g., "System

---

12 [AFFAARS] *AFFARS 5315.304. Evaluation factors and significant subfactors,* "(b): It is Air Force policy to establish the absolute minimum number of factors necessary for evaluation of proposals. Source selection factors may be subdivided into subfactors that, in rare instances, may be further subdivided into elements if needed for Agency source selections. Evaluation factors and, if used, subfactors and elements, are the basis for assessing each offeror's ability to meet the Government's needs. ... (c): Source selections shall use the following four evaluation factors: Cost or Price, Past Performance, Mission Capability, and Proposal Risk. For Basic source selections, however, evaluation of proposal risk is optional. The Mission Capability factor shall be limited to six subfactors, unless additional subfactors are justified, documented in the SSP, and approved by the SSA. Proposal risk shall be assessed at the Mission Capability subfactor level. Subfactors are not normally used for Past Performance and Cost or Price."

13 e.g., the questions and criteria found in [Haddad 1998a] (op. cit), and Appendices C and D of this document.

The Offeror shall provide an overall description of the software development effort for the offeror and all *software team members*[1]. This description shall include:

a) an integrated <***program name***> Software Development Plan (SDP) and
b) the response to the Software Development Capability Evaluation (SDCE) as described in Annex <*X*> to this RFP along with substantiating documents.

The SDCE response shall be a unified response from the Offeror and software team members with *significant software responsibility*[2]. The response shall indicate that the Offeror's and *significant software team members*'[3] software development processes, standards, policies, methodologies, tools, technologies and facilities are in place and are adequate to provide a sound, disciplined, systematic, and managed approach to software development and sustainment. The integrated <***program name***> SDP shall be a combined SDP for all the significant software team members. See Annex <*X*> for specific instructions for the question responses for the SDCE. (SDP and SDCE responses are submitted in Other Contract Documentation.)

[1] A *software team member* is any internal or external organization that develops, tests, or supports software-related work being performed for this contract and has an agreement (formal or informal) with the prime contractor or any subcontractor. These organizations include, but are not limited to, intra-corporation software organizations, in-house service providers, developers, fabrication/manufacturing organizations, laboratories, and subcontractors. Examples of an agreement include a contract, work authorization, memorandum of agreement, or oral agreement.
[2] *Significant software responsibility* includes responsibility for any deliverable software (including the software portion of firmware) or for any software used in satisfying, verifying or validating requirements or used in performing or supporting operations or sustainment (e.g., applications, security, safety, training, simulation, analysis, database support, automatic test equipment, maintenance).
[3] A *significant software team member* is a software team member with significant software responsibility.

Figure 3-2. Sample language for Section L—Instructions to Offerors.

This requirement has been met when:

a) The Offeror's and their software team members' unified SDCE responses:
   1. Satisfy all criteria specified in Annex <*X*>;
   2. Define processes, standards, policies, methodologies, tools, technologies and facilities that the Offeror has demonstrated can be used effectively in a sound, disciplined, systematic, and managed approach to developing software that satisfies the <***program name***> program schedule, performance, quality and sustainment requirements.

b) The Offeror's and their software team members' unified SDCE responses are consistent with the contents of their SDP and the IMP, Integrated Master Schedule (IMS), Work Breakdown Structure (WBS), and SOW.

c) The Offeror's and their software team members' software development processes, standards, policies, methodologies, tools, technologies, facilities, and management plan(s), as specified in the IMP and SDP, can be used effectively in a sound, disciplined, systematic, and managed approach to developing <***program name***> software that satisfies schedule, performance, quality and sustainment requirements.

Figure 3-3. Sample language for Section M—Evaluation Criteria.

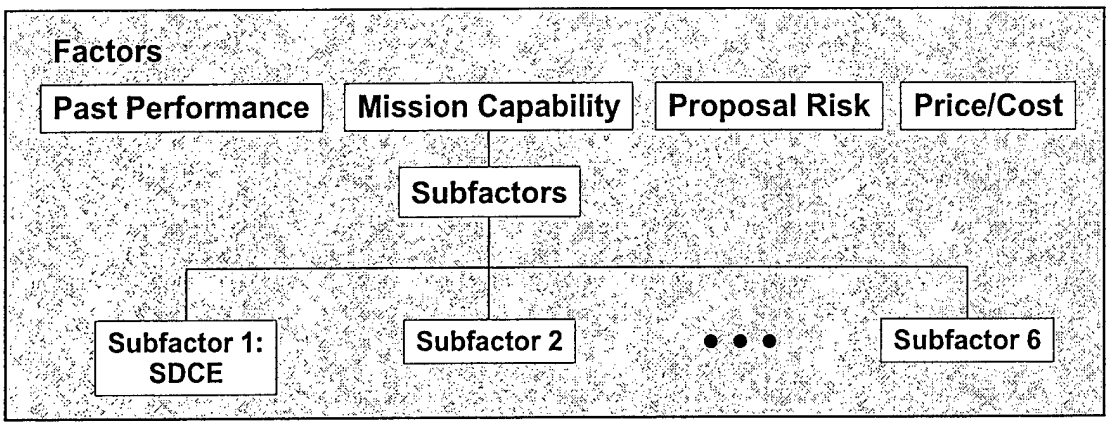


Figure 3-4. Recommended placement of SDCE in RFP structure.

Engineering," "Software Development." This case diminishes the impact of the SDCE due to "roll ups" with other elements under that subfactor and is, therefore, less desirable. In any RFP, the case where the SDCE questionnaire is divided among the subfactors (see Figure 3-5) must be avoided; this approach will hinder the ability to perform consistency checking between subfactors and their SDCE responses and eliminates any impact the SDCE as a whole would have on the source selection results.

The SDCE may also be performed outside of a source selection if the acquisition strategy has adequately addressed its use. The concept is to perform the SDCE as part of the current contract phase to support source selection for the next (following) contract phase. An example of this approach is depicted in Figure 3-6. In this case, the program's SDCE is performed during an early acquisition life-cycle phase where multiple, parallel, competing contracts are issued, prior to a downselect to fewer contractors (or one single contractor) after the source selection for the next acquisition phase. The results of the current-phase SDCE are included as part of the selection process during the next-phase source selection.

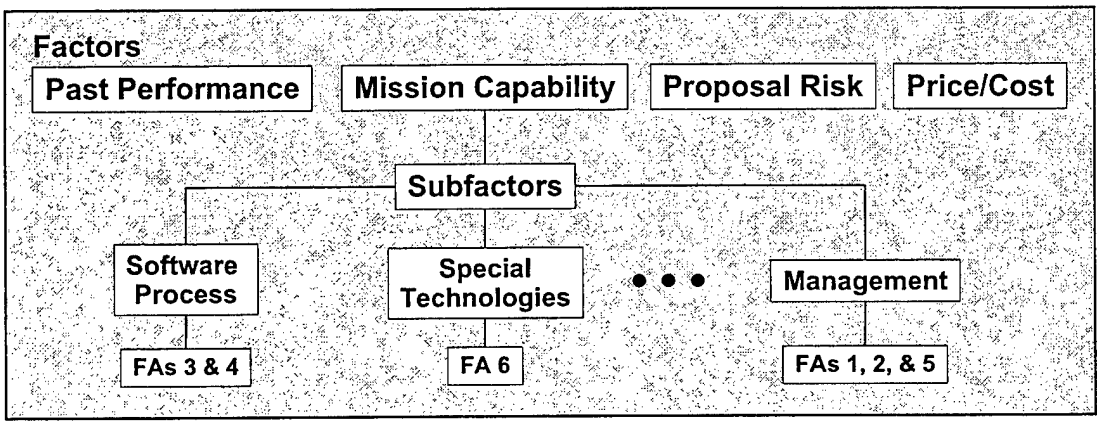


Figure 3-5. Poor placement of SDCE in RFP structure.

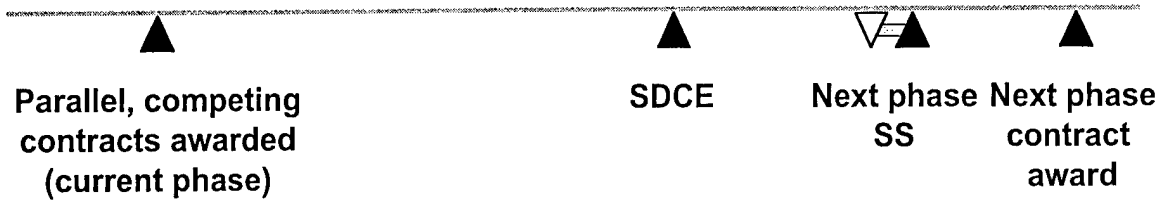


Figure 3-6. SDCE outside of source selection (SS).

This type of SDCE is done for several reasons. First, a current-phase SDCE does not have the same degree of time constraint as an SDCE done during source selection. The streamlined source selection process only allows for a limited time to evaluate each offer—in general, about two to three weeks. It may be impossible for a program office to obtain enough trained staff to complete even a moderately sized SDCE (60–80 questions) in this limited time. However, a current-phase SDCE, completed prior to the following acquisition phase, allows more time for evaluation and takes fewer physical resources to complete. Second, the SDCE can help identify risks in the current phase of the contract, allowing the contractors the opportunity to improve their processes and implement risk mitigation efforts prior to the start of the following phase. Finally, the detailed SDCE results identify process weaknesses that are shared with the contractors for internal process improvement.

In order to support this approach, the program office must include contractual language in both the preceding-phase RFP and the following-phase RFP to perform an SDCE in mid-phase. For the preceding phase, the RFP language should require the contractors to prepare for an SDCE during that phase and to include the resources for performing an SDCE in their bid. For the following phase, the RFP evaluation criteria must include the results of the SDCE performed during the preceding phase. However, it will be necessary to perform an additional evaluation of the contractors' software development processes during the following-phase source selection since the proposal, Integrated Master Plan (IMP), Integrated Master Schedule (IMS), and final SDPs for the following-phase contract are not available to the SDCE team in mid-phase.

Although the current-phase SDCE relieves the program office of some time constraints and aids in process improvement activities for each contractor, it is critical to obtain early approval from the Project Contracting Officer (PCO) to perform the mid-phase SDCE before model and process tailoring begins. Without written approval from the PCO, the SDCE may be moved into the following-phase source selection, which impacts both resources for performing the SDCE and time allocation to provide an adequate evaluation. The program office should also provide incentives to contractors in their award fee criteria that motivate improving their software processes based on the SDCE results.

3.1.6 Select and Prepare Team

As the cost, schedule, and tailored questionnaire are coordinated, the SDCE team leader must assemble an experienced software evaluation team. These team members should meet the qualifications for participation on an SDCE (AFMCPAM 63-103, Volume 1, Section 4.B.3) and be capable of fulfilling the entire source selection time commitment. As a goal, the SDCE team leader should mentor at least one team member for qualification as an SDCE team leader, and a less-experienced software engineer to qualify as a team member. If possible, the evaluation team should include Government as well as

FFRDC (e.g., Aerospace) software specialists. The team leader and team members must also satisfy all requirements of the source selection (e.g., conflict of interest rules, non-disclosure agreements).

Once the team is established and management commitments to the schedule are obtained, the SDCE team leader provides training on the SDCE process, site visit performance, and development of final evaluation results. Each team member will also take part in source selection training to ensure that each person understands the legal implications of participating in the evaluation and the processes and tools to be used for the source selection. Once all SDCE team members are trained and briefed into the source selection, they may review the RFP and other Government source selection documents to familiarize themselves with the program. The activity of selecting and preparing the team may continue after release of the RFP but must be completed before the start of the source selection evaluation process.

If questions and criteria from the special technologies addressed under Functional Area 6 are part of the questionnaire, personnel with expertise in those technologies must be included on the SDCE team. The experts may not be required for the complete SDCE schedule, but must be trained for both the source selection and SDCE, and must be included in the SDCE budget.

3.2 Conduct Evaluation

3.2.1 Review Proposals/Offeror Responses to Questionnaire

During the source selection, the SDCE team will evaluate each offeror's response to the questionnaire. This response consists of a short (approximately two page) essay answer to each question in the SDCE questionnaire, and should relate to the criteria associated with each question. Additionally, the offeror must provide direct evidence of use of the process covered in the short essay answer; this evidence should be drawn from a previous phase of the current program or from a program deemed similar to the program under bid. Each piece of evidence is accompanied by a Cover Sheet, which describes how the evidence relates to the program under bid. A modified reproduction of the cover sheet from AFMCPAM 63-103, Volume 2, Attachment 3-3, is provided in Figure 3-7. A completed example of the cover sheet is provided in Figure 3-8. The SDCE team uses the cover sheets as a "first look" at the evidence of use provided by the offerors. The evaluators should note those areas where the evidence does not meet the criterion for similarity to the program under bid, e.g., application domains are widely disparate or the programming languages are not related (e.g., high level vs. low level). The cover sheet contents also suggest areas of weakness or strength to the evaluators where they may delve more deeply into the offeror's processes and evidence to substantiate their initial impressions.

The SDCE schedule will generally start on the day of proposal receipt or the following business day. The SDCE team, using the process established by the SDCE team leader, will review each offeror's proposal and the SDCE responses. The evidence of process use associated with each of the SDCE responses is assessed as well, to ensure that the offeror has practical experience in implementing the given process and that the process is adequate for performing successful software development. The response-to-evidence cross-reference matrix, required under the SDCE instructions (see Sections 5 and 6 for example instructions), facilitates the evidence review. The evaluators coordinate information in the IMP, IMS, and proposal with the response and evidence of use, noting those areas where

Cover Sheet for Project's Sample Data		
Contractor:		
Sample Project Name:		
Sample Project Customer:		
Critical Capability(ies):		
Title of Sample:		
Explain why your experience on the sample project is relevant to the proposed project.		
ATTRIBUTES	**PROPOSED PROJECT**	**SAMPLE PROJECT**
Application Domain		
Product Type		
Acquisition Phase[1]		
Software Development Phase(s)		
Award Date		
Contract Duration		
Current Project Phase/ Contract Month[2]		
Prime/Subcontractors[3]		
Software KSLOC[4]		
Language(s) and Percentages		
Target Processor(s)/OS(s)		
Applicable Standards		

[1]For "Proposed Project," phase(s) in which Critical Capability(ies) are to be used; for "Sample Project," phase in which sample was generated.

[2]Phase/month of the Sample Project as of the current date.

[3]Contractors developing the software products specified in the "Product Type" row

[4]Total number of KSLOC for software specified in the "Product Type" row

Figure 3-7. Cover sheet for SDCE evidence of use.

Cover Sheet for Contract's Sample Data		
Contractor: Team A; Rolling Hills, VT		
Sample Project Name: Project X		
Sample Project Customer: U.S. Air Force Space and Missile Systems Center		
Critical Capability(ies): 4.4.2 Metrics Application		
Title of Sample: Project X Software Development Metrics Reports		
Explain why your experience on the sample project is relevant to the proposed contract.: Object-oriented methods and metrics were used on the sample project. The same object-oriented methods and metrics are planned for use on the proposed contract.		
ATTRIBUTES	**PROPOSED CONTRACT**	**SAMPLE PROJECT**
Application Domain	Weather Satellite	Communications Satellite
Product Type	Ground System (Command and Control)	Ground System (Command and Control)
Acquisition Phase[1]	EMD	EMD
Software Development Phase(s)	Design; Coding and Unit Test	Design; Coding and Unit Test, Increments 1 and 2
Award Date		1/94
Contract Duration	8 Years	5 Years
Current Project Phase/ Contract Month[2]		EMD: Between System PDR and System CDR/Month 24
Prime/Subcontractors[3]	2 Software Subs	Prime & 1 Software Sub
Software KSLOC[4]	750	500
Language(s) and Percentages	Ada 95: 90%; C++: 10%	Ada 83: 75%; C++: 25 %
Target Processor(s)/OS(s)	RISC 6000/UNIX	VAX 6200/VMS 6.2
Applicable Standards	EIA/IEEE J-STD-016-1995	DoD-STD-2167A & 2168

Figure 3-8. Sample completed cover sheet for SDCE evidence of use.

the offeror's software development processes exceed, meet, or fall short of the requirements of the RFP and SDCE question/criterion. These notations form the initial findings for the SDCE, and, based on the SDCE team leader's process, will be used to establish the final findings for the SDCE and the evaluation against the Section M criteria. The SDCE team will review one offeror team's submission separately before reviewing the submission from another team. Equal time periods are spent on each offeror team.

3.2.2 Prepare Evaluation Notices (ENs)

Based on the SDCE team findings, and in order to solicit additional information from the offerors, the SDCE team will generate ENs to be submitted during the discussion phase of source selection. The ENs are used to provide questions on discrepancies in the proposal material, IMP, IMS, SDCE responses or evidence of use, or note risk areas in the offerors' proposals that could materially affect program success. The ENs are generally developed and submitted to the source selection team leads on-line, using a source selection tool. They should provide sufficient information to the team leaders (and hence, the offeror) as to where the problem was noted, provide the request for additional information (either as a question or statement), and state the impact of the unresolved problem or risk to the program. This information is used by the source selection team leaders (subfactor lead, factor lead, team leader, and PCO) to determine whether the requested information provides a software discriminator and warrants a request to the offerors. Therefore, it is imperative on the SDCE team to coordinate their findings prior to generating their ENs to ensure that their questions and risks identify the significant software issues.

3.2.3 Perform Site Visits (optional)

If allowed, site visits are performed after the SDCE team completes their review of the proposal material, SDCE responses, and evidence of use. The program office can include the intent to perform a site visit as part of the RFP and SDCE instructions, even if a site visit is not later performed, to allow the offerors time to prepare facilities and resources for discussions. Once the determination to perform a site visit has been made, the PCO will send a formal notification to each offeror stating the intent to perform the visit, the schedule for the visit, and the agenda. Discussion topics and ENs to be covered during the visit are sent under cover letter with the agenda. The SDCE team leader should coordinate with the program office and the evaluation team to ensure that the team's concerns are allocated adequate time during the visit. The SDCE team leader should also provide site visit training so that all included team members are aware of the limitations on discussion topics, the focus of the submitted ENs, and the areas to be investigated during the visit (e.g., development labs, test facilities). The PCO may also attend to ensure that no improper communication takes place between the SDCE team and the offeror.

During the site visit, the SDCE team members should elicit as much information as possible regarding the EN responses and follow-on questions without leading the offerors to infer specific technical solutions. Questions should be phrased to understand "how" the offeror plans to perform tasks, and not recommend an approach. The SDCE team also cannot discuss the SDCE results, SDCE and source selection color or risk ratings, comparisons between offerors, whether the offeror meets the evaluation criteria, or offer value judgments on information uncovered during the site visit. The site visit should be considered a fact-finding mission and not a technical interchange meeting.

Once the site visit is concluded, the SDCE team should provide feedback to the offerors on the results of the visit. The team should convey their observations without relating positive or negative judgments from those observations. The feedback session also provides the offerors a last chance to respond to questions and observations.

3.2.4 Analyze EN Responses

EN responses are formally submitted by the offerors to the SDCE team through the PCO. Team members responsible for the EN question must review the offeror's response and any materials that may have been submitted with the response, such as additional evidence of use of a particular process. If the response provides adequate information to answer a question or area of concern, the team member will document the analysis results in the EN and submit it for closure. If the offeror's response is deemed inadequate, the team members may want to provide another EN to cover the remaining questions, as long as discussions allow, or document the question as a proposal risk to be raised if that offeror is selected as the winning bidder. At some point, the SDCE team leader may determine that submittal of ENs has reached a point of diminishing returns and document the issues as proposal inadequacies and/or risks.

3.2.5 Establish SDCE Results (Determine strengths, weaknesses, and risks)

Once the EN responses have been finalized and evaluated, the team's findings related to each SDCE question response, applicable substantiating evidence, and related proposal information form the basis for the SDCE results. The SDCE team leader and team members use the findings to perform an overall evaluation of the offeror's software development approach, generally by consensus.

The team should not "roll up" the findings for each question under a CC into a top-level Functional Area result or color. This tactic dilutes the potential of the SDCE to reveal specific strengths, weaknesses, and risks in an offeror's software capability. Instead, the team should discuss the findings to determine whether there are overarching strengths or problems in an offeror's process or its implementation. The findings may cut across several CCs, CCAs, and FAs, and should be coordinated and used to determine whether the offeror has exceeded, met, or failed to meet the criteria for software development processes as stated in Section M of the RFP (see Figure 3-3). The overall strengths, inadequacies, and risks are documented and submitted to the subfactor and factor leads for further discussion with the offeror, or for inclusion in the final roll up of the subfactor.

3.2.6 Integrate with Source Selection

Once the SDCE team has completed their assessment and determined the level to which the offeror has met the Section M criteria, the SDCE team leader will work with the responsible Section M evaluator or Mission Capability subfactor lead to establish the color rating (Blue, Green, Yellow, Red) for the criteria and perform the proposal risk assessment (High, Medium, Low) relative to the offeror's software capability. The colors, associated ratings, and definitions are described in Table 5315-3 of the AFFARS and are used to establish the level to which the offeror meets minimum performance or capability requirements. Proposal risk ratings are defined in Table 5315-4 of the AFFARS, and focus on the proposal weaknesses, where they may impact schedule, increase costs, degrade performance, or increase the need for government oversight.

Section M criteria related to the SDCE may form elements of a subfactor under the Mission Capability factor. In this instance, a "Green" (acceptable) or "Yellow" (marginal) color rating may have little or no impact on the source selection results. If the SDCE is a subfactor in the RFP, the color rating is reported to the Source Selection Authority (SSA) as it stands since subfactors under the Mission Capability factor are not rolled up to a higher level. This increases the visibility of process strengths, weaknesses, and risks to the source selection evaluation team (SSET) and source selection authority (SSA). In either case—element or subfactor—"Blue" or "Red" ratings have a high impact on the source selection results. A "Blue" (exceptional) result adds a strength to the overall proposal, indicating a material benefit to the government if the offeror is selected. Conversely, a "Red" result makes any proposal unawardable, since this indicates that the offeror has failed to meet a material requirement imposed by the customer.

The SDCE team leader will also document the proposal risks associated with the SDCE Section M criteria. These will be coordinated with the Section M evaluator or subfactor chief since these will become part of the formal source selection materials. As stated in AFFARS 5315.305(a)(3)(B), a combination of significant proposal risks may result in an unacceptable level of risk and a proposal deficiency.

Based on the acquisition strategy, the program may allow for final updates to proposals from each offeror (referred to as Final Proposal Revisions (FPRs)). The FPRs are submitted after the PCO determines which of the offerors meet the competitive range decision criteria. The evaluation of FPRs is similar to the original evaluation task, but the SDCE team will evaluate only the SDCE-relevant portions that have been modified by the offerors. In general, the team will not re-evaluate originally submitted responses and data. Formal ENs may be submitted after FPR review, but due to time constraints, the PCO may instead allow a short (e.g., one week) round of clarification discussions with each offeror prior to the contract award decision. The SDCE team should be prepared to assist the PCO and subfactor lead with clarification questions and response evaluations as part of the FPR review process. The steps for completing the final evaluation are the same as the steps cited for establishing the initial proposal SDCE results and pre-FPR subfactor evaluation results.

3.3 Report Evaluation Results

3.3.1 Conduct Feedback (optional)

After the SDCE has been completed, the SDCE team should attempt to present feedback on their findings to each offeror (permission to hold this type of discussion is at the behest of the PCO and should be requested by the SDCE team leader prior to the start of the source selection). This gives each offeror an opportunity to see details of their evaluation—where the SDCE team found specific strengths and weaknesses in the offeror's processes or in the evidence of use. This type of feedback should emphasize where the offeror might focus internal process improvement efforts and/or highlight lessons learned for their software engineering process group (SEPG).

However, these feedback sessions rarely happen. In past source selections, the SDCE team leader did not attend the offeror outbriefings since the PCO or SSA usually held these before a limited audience. In the nominal case, once EN response evaluation is completed, discussions closed by the PCO, and the contract award determined, the SDCE team summarizes their findings in briefing bullets for pres-

entation to the losing bidders and to the winning contractor. Process strengths and inadequacies not addressed by the EN responses are submitted to the responsible evaluator or subfactor chief for inclusion in each offeror's outbrief. The team should attempt to convey to the subfactor and factor leads the need to instruct the offerors on their process inadequacies, which may be forwarded to the offerors in their respective outbriefs. But the outbrief takes a more usual form of general concept bullets, based on strengths and inadequacies that have been abridged by the SSET; these provide a minimum level of feedback to the offeror, but are not sufficient for process improvement. Additionally, proposal risks should be briefed, not only for the understanding of the winning offeror, but also to highlight these areas to the program office as "watch list" items or initial risks that should be monitored through program execution.

3.3.2 Transition SDCE Results

Following the announcement of contract award, the SDCE findings should be conveyed to the program office for their use to monitor risk areas and to ensure that any inadequacies in the winning offeror's processes are tracked for improvement activities. At this point, the SDCE team has completed their tasks associated with the evaluation. The program office, however, should consider including at least one of the SDCE team members as part of their software staff to monitor the contractor's process use and improvement. The SDCE results, as part of the source selection materials, have not been transitioned in the past due to restrictions on viewing source selection sensitive information. Recent discussions with PCOs on several programs provide hope that these restrictions may be lifted or waived for the SDCE results so that both the contractor and program office may benefit from them.

3.3.3 Program Follow-Through

The program office must be vigilant throughout the program lifecycle to ensure that the contractor's software processes are followed and enforced. As a program progresses, the contractor may relax or deliberately abandon any software processes that are perceived to affect the development schedule adversely or cause cost overruns. To avoid or uncover software development problems after contract award, the program office may request a "mini-" capability evaluation to assess the ongoing software development capabilities of the contractor. Perceived risks and inadequacies may be assessed using a small subset of the SDCE questionnaire, with substantiating evidence provided only from the current project. The program office uses results of the evaluation to identify risk areas in software development, which should be tracked and managed as part of the program's overall risk management process. One caveat of this approach is that any capability evaluation will take time and effort away from the current project and should only be performed with discretion. If possible, the requirement for a government mid-term assessment should be part of the RFP to allow each offeror to bid the effort and to schedule resources to participate in the assessment. To preclude a large effort for a mid-term assessment on the contractor's part, one NRO program sent a small SDCE questionnaire to the contractor, which entails only project-related evidence and oral responses to the questions during a three-day site visit—no written responses or off-project documentation were required.

4. The Basic Software Development Capability Evaluation

As stated in earlier sections of this document, SDCEs have been performed for 25 SMC and NRO programs, with the team members consisting primarily of personnel from program offices and Aerospace. The SDCE-experienced personnel are a limited resource with many demands on their time. Additionally, acquisition reform has reduced the size of program office and Aerospace staff, leading to constrained program office budgets. Unless the Level 3-equivalent SDCE is required by the program office, the authors of this document, along with other experienced SDCE staff, recommend performing a smaller, tailored SDCE during source selection that focuses on identifying the program's software risks and meets the program's schedule, budget, and resource requirements. This basic SDCE is provided in Appendix A.

4.1 Background

For the known SDCEs that have been performed for SMC and the NRO, the number of questions selected for an evaluation range from a high of 550 to a low of 11 on the EDS 2001 source selection, yielding an average of 83 questions per SDCE for those with known numbers.

Table 4-1 provides additional information on these SDCEs.

Table 4-1. SDCE Statistics for SMC and NRO

Program	Date	# of Questions	Site Visit
DMSP STT	FY94	Unknown	Yes
ACMS	FY94	Unknown	Yes
DMSP CDFSII	FY94	100	Yes
GPS OCS	FY95	118	No
RSA Phase II	FY95	72	No
GPS Block IIF	FY95	27	No
AFSCN RCDC	FY95	32	No
AFSCN NOUC	FY95	79	No
AFSCN CCSC	FY95	48	No
SBIRS High	FY95	98	No
Classified	FY96	50	No
ABL	FY96	550[14]	No
EELV Pre-EMD	FY96	36	No
GBS	FY97	66	No

14 Generated and performed by a team with limited training.

Program	Date	# of Questions	Site Visit
EELV EMD	FY98	68	No
NPOESS OMPS	FY99	26	No
NPOESS CrIS	FY99	26	No
SLRS	FY00	22	No
EDS 2001	FY00	11[15]	No
WGS	FY00	37	No
MILSATCOM (CCS-C)	FY00	36	No
SBIRS Low	FY01	210[16]	No
AEHF	FY01	130[17]	No
SCNC	FY02	32	No
GPS DAGR	FY02	28	No

After the first nine SDCEs were completed, the SDCE focal point at that time developed a list of "most frequently used" questions,[18] most of which have been carried over in one form or another to the current basic SDCE. In the nine years that have followed the instantiation of the SDCE, program office concerns regarding software technology and software costs have changed, and the basic questionnaire has been augmented to alleviate some of those concerns, specifically in areas of program planning and tracking, risk management, software quality, and defect control. The use of the basic SDCE questionnaire (given in Appendix A) as a baseline provides the program office and SDCE team leader with a starting point for risk identification and mitigation, and software process discriminators.

4.2 Tailoring/Augmenting the Basic SDCE

The entire (updated) SDCE consists of over 700 questions and criteria related to every aspect of software development. Primarily, the SDCE's CC structure provides at least one high-level question for each CC, followed by lower level questions/criteria designed to probe into the offeror's understanding of the CC's topic. The basic SDCE discussed here contains only 41 questions, at both high and low levels, which address portions of program management, systems and software engineering, commercial software selection and deployment, software quality, and development environments. These questions were selected for two reasons: first, they provide coverage of most of the software problems/risks that are inherent in currently fielded or failed systems, and second, the small number allows for completion of the evaluation in a source selection timeframe using a minimum of resources.

---

15 The EDS SDCE questionnaire consisted of questions and criteria related to the selection, integration and maintenance of Commercial Off-the-Shelf (COTS) software products. A set of COTS questions and criteria is included in Appendix D of this document.

16 The SBIRS (Space-Based Infrared Systems) Low SDCE was the first Level 3-equivalent SDCE performed in support of a contract award. The question set consisted of 158 questions from Functional Areas (FAs) 1-5 and 52 from FA 6. Section 5 of this document provides more detail on the Level 3-equivalent SDCE.

17 Tailored Level 3-equivalent SDCE.

18 [Haddad 1998b] Haddad, R. W., *Guidelines for the Use of SDCE*, The Aerospace Corporation TR-98(8550)-2, 1 March 1998, Section 6.

However small the basic SDCE questionnaire may be, there may be times when a program's budget or schedule precludes the use of all 41 questions, or the program's risks may be more focused on a particular area. In these instances, the program office and SDCE lead have choices to make: if budget and schedule allow only for a shortened SDCE, and resources are available, the SDCE team leader and his/her program office counterpart jointly determine where the small SDCE is excessive and tailor out or change questions and criteria appropriately. Once the tailored questionnaire is completed, the question set and instructions are provided to the PCO and other staff members for review. If there are no issues or modifications, the SDCE instructions and questionnaire are included in the RFP.

Conversely, if the program office determines that the small SDCE does not provide adequate coverage of the identified risks, the SDCE team leader and his/her counterpart should map the program's risk list to the small SDCE, noting areas where coverage is limited or nonexistent. From there, the remaining risks are aligned with the entire SDCE, top down from FAs and CCAs, to CCs and questions/criteria. The program office and SDCE team leaders may select additional questions (or modify existing questions and criteria, as needed) from the SDCE pamphlet and other sources ([Haddad 1998a] and Appendixes C and D of this document) that encompass their risk areas. This is an iterative process of selecting, reviewing, and accepting or rejecting that requires sufficient knowledge of software systems engineering and management in order to select appropriate questions and criteria for augmenting the small SDCE. After completing the extended SDCE, it is given to the program office staff and PCO for review and comment; after the review is complete, the final questionnaire and instructions are given to the PCO for inclusion in the RFP.

4.3 Resources Required

A small SDCE, similar to the one provided in Appendix A, can be performed by a trained SDCE team of two or three full-time members during a "normal" source selection, where a two-to-three-week period is scheduled to complete the evaluation of each offeror's proposal. The average amount of time required for an evaluator to complete an assessment of an offeror's response to a specific question is estimated at four hours, based on experience and SDCE team feedback. This allows time for reading the question response and the substantiating documentation, the relevant parts of the proposal, other proposal documentation (e.g., the IMP), and documenting the findings. The small SDCE with 41 questions would take about two weeks for a 3-member team to complete, with each team member reviewing and documenting findings on 1/3 of the responses and reading through the other 2/3 of the questionnaire as a sanity check for the rest of the team.

However, there is no simple algorithm for estimating the level of support for all SDCEs. Here we provide some examples of the resources required from the SDCEs performed at SMC:

- The EELV EMD SDCE, done ahead of a downselect source selection, had 68 questions and required a team of nine evaluators (two full-time, the rest part-time, with one trainee) one month to assess two prime contractors and their subcontractors. At least two team members reviewed each question; results were determined by team consensus.

- The SBIRS High SDCE contained 98 questions and was performed by a team of nine evaluators during source selection, with two weeks allocated for each offeror. At least

two evaluators reviewed each response, requiring extended work hours each business day and weekend work as well.

- Teams of two members each worked three small SDCEs: SCNC, SLRS, and EDS 2001. Their efforts were completed in one week for each offeror on EDS 2001, and two weeks for each offeror on SCNC and SLRS.

- The SBIRS Low SDCE was also done prior to the down-select source selection. It required a staff of five full-time and five part-time evaluators, and took over seven months to complete. This was the pilot for the Level 3-equivalent SDCE; its plan required that each response be reviewed by at least two team members, and consensus meetings were held to ensure that all findings were coordinated and documented across the entire questionnaire. The major differences between this evaluation and other large SDCEs were: (1) the SBIRS Low evaluators had to document their reasons for accepting or rejecting the response to each of the questions/criteria, and (2) the evaluation team provided a detailed final report to each of the contractors for their use in process improvement.

Based on these examples, one can conclude that a small SDCE can be done in a short period of time by a small, trained team, but that larger SDCEs can also be done in a short time, depending upon the number of evaluators and the plan for performing the SDCE (how the evaluation is being done, consensus vs. individual findings, final report generation, etc.). Given the limited resources at SMC for performing SDCEs, any large evaluation is difficult to staff and to complete within the shortened source selection timeframe.

4.4 Benefits of the Basic SDCE

The Basic SDCE has several benefits for both the program office and the offerors. It requires fewer evaluators to perform the assessment and determine results, allowing for a reduced source selection budget or allocation of additional personnel to higher risk evaluation areas. A small team can assess their findings and generate evaluation notices more quickly than a large team, giving the PCO and subfactor lead more time to assess the findings, submit their ENs, and receive responses sooner from the offeror—all of which aid in keeping the source selection on schedule. It focuses the program office on the key risk factors in their solicitation, without attempting to uncover every issue related to software development. With fewer, but more focused, questions, the responses to each question can become true discriminators between offerors for the program's source selection, which also supports the requirements of the AFFARS.[19] Finally, the level of effort required on the part of the offeror is reduced since they have fewer questions to which they must respond and provide substantiating documentation.

19 [AFFARS] AFFARS 5315.304 (b). "Evaluation factors, subfactors, and elements: (ii) Shall include only those specific program characteristics that are tied to warrior needs, significant enough to have an impact on the source selection decision, and expected to be discriminators based on market research"

4.5 Disadvantages of the Basic SDCE

Although the Basic SDCE provides some distinct advantages for the program office and offeror, in its reduced level of effort to perform and its focus on major risk areas, it can be problematic in other areas. In their zeal to keep to a resource-limited, short source selection timeline, the program office and SDCE team leader may trim the small SDCE even further, inadvertently eliminating questions that could elicit advantageous information on processes in use by an offeror. Conversely, they may also eliminate questions that could highlight deficiencies in an offeror's processes. If the SDCE team leader finds that the program office wants to drastically cut the questionnaire, he/she should reiterate the reasons for performing the SDCE (uncover risks, provide discriminators, assess processes) and spell out the risks that are incurred by eliminating questions.

5. The Large Software Development Capability Evaluation

5.1 Background

An October 1999 DoD policy statement required that for contract award, offerors on ACAT I and IA contracts must be evaluated at Capability Maturity Model (CMM) for Software (SW-CMM) Level 3, or its equivalent level using a DoD-approved tool. Through mid-2000, the Software Intensive Systems Working Group for Level 3-equivalent methods, a team of industry, military, government and Federally Funded Research and Development Center (FFRDC) personnel, assessed the SW-CMM Level 3 methodology and its correlation to the SDCE. The outcome of the assessment was a DoD-approved SW-CMM Level 3-equivalent SDCE core questionnaire, consisting of 130 questions and related criteria that may be applied on ACAT I and IA programs undergoing source selection. Additionally, the team developed a set of requirements for process evaluation methods, which would have to be satisfied by any evaluation method prior to its acceptance as a DoD-approved tool.[20]

The Level 3-equivalent, or "large," SDCE has been performed only once on an SMC ACAT I acquisition to meet the intent of the DoD policy. During FY03, the policy for ACAT I and IA programs was reduced from mandatory to guidance. In addition, new acquisition policy for space programs was released that also reduced the policy to guidance for space programs. The following sections contain the lessons learned from the pilot Level 3-equivalent SDCE. These lessons apply to any SDCE of a similar size.

5.2 The Large SDCE Pilot

5.2.1 Description

The large SDCE was performed for a software-intensive satellite acquisition after the program had passed the Program Development/Risk Reduction (PD/RR) acquisition milestone and was moving toward a downselect to a single contractor for the Engineering and Manufacturing Development (EMD) phase. Since the large SDCE had not been performed for any other acquisition, this program became the pilot program for the Level 3-equivalent SDCE.

5.2.2 Questions and Criteria

The set of SDCE core questions and criteria was developed by a team of Department of Defense (DoD), industry, and FFRDC staff who were familiar with the SDCE and the SCE, and had been participating on the Software Intensive Systems Working Group for Level 3-equivalent methods. The direction from Undersecretary of Defense for Acquisition, Technology and Logistics [USD (AT&L)] was to reach an "80% agreement" on the mapping of SDCE questions and criteria to the SW-CMM goals, capabilities, activities, abilities, and measurements for the Level 2 and Level 3 KPAs. The

20 [DoD-SEIPT 2001] *The Requirements for Process Evaluation Methods and Their Application*; DoD Software Evaluation IPT; 11 April 2001 (unpublished).

mapping team was restricted to the language of the AFMC pamphlet, which contains some inconsistencies in contents and obsolescent references to standards and practices; however, they were allowed to add entirely new questions and criteria in order to address the Level 3 assessment practices not specifically covered by the SDCE (e.g., institutionalization of processes). Appendix C contains the SDCE core criteria, questions, and their mappings to the SW-CMM key practices.

The SDCE questionnaire for the pilot program also included questions and criteria that were not part of the core question set. Most of these questions were from FA 6, Program Specific Technologies, and focused on artificial intelligence, trusted systems, distributed network-based systems, and open systems. Other topics not covered by the core question set, but considered to be program-specific risks and added to the pilot program SDCE, were: systems and software engineering integration; system/software architecture definition; incremental software development; integration and qualification test resources/facilities; and COTS and reuse software. Overall, the team evaluated responses to over 300 question/criterion pairs.

5.2.3 Contractors' Instructions for Completing the Level 3-Equivalent SDCE

The instructions for the Level 3-equivalent SDCE are presented in Appendix B, and are similar to those for the contractors on the pilot project. As with the instructions for the basic SDCE, this instruction set provides explicit direction to the prime contractors and their significant software team members on presentation of their development organization, team responses, and substantiating documentation. However, due to the performance of the SDCE outside of source selection, the instructions must include the proposal submission requirements normally specified in a Request for Proposal (RFP). Proposal submission requirements include instructions for formatting the proposal (e.g., font sizes and margins), electronic submittal requirements (e.g., applications used for generation, virus scanning requirements, and delivery method), volume layout and referencing (such as binder size and information tabs), and other information (e.g., page limits and submittal dates). Since the results of the SDCE are to be included as part of the next-phase source selection, the Level 3-equivalent SDCE must also conform to the requirements of that source selection. This differs significantly from the basic SDCE instructions since the basic SDCE instructions are included as an annex to the RFP and are covered by the general instructions of the RFP. These additional instructions apply to any SDCE being performed during the contract period (i.e., outside of source selection).

5.3 Evaluation Team Training

In order to accomplish the SDCE and provide results on a criterion-by-criterion basis, similar to the results produced by a key practice-oriented SCE, the SDCE team leaders developed specific training for completing the evaluation. In the training, the team leader noted that, although the offerors provide responses to each question, the evaluators would evaluate by criterion, with all of the associated questions for that criterion. Hence, the evaluators were required to read all contractor responses to all questions associated with a criterion, and review the substantiating evidence for those responses, prior to making an evaluation of the criterion. The team also developed new forms to document the evaluation of each core criterion, which could be used to verify whether the offerors had satisfied the requirement for "equivalence" to SW-CMM Level 3—if all core criteria were satisfied, the offeror's processes and application of those processes were deemed to be equivalent to those of a SW-CMM Level 3 organization.

Since the SDCE was expected to be used as input to the following-phase source selection, the team leader provided training on the source selection FAR/AFFARS (Air Force Federal Acquisition Regulation Supplement) terminology, which defined "evaluation notices" and the significance of "strength," "inadequacy," "deficiency," and "risk" to the evaluators. Additionally, the team was trained to report "observations," which has specific meaning with regards to the SCE. The team also received a detailed review of the instructions to the offerors on completing the SDCE (see Appendix B.2), which were significantly expanded over those used for the basic SDCE (provided in Appendix A).

A portion of the training for the large SDCE consisted of the basic SDCE training on the methodology and structure of the questions and criteria. Other training areas covered specific teaming arrangements and responsibilities, development of required documentation (e.g., evaluation notices and risk assessments), and consensus building. Additionally, since the core set of questions and criteria cannot be tailored, obsolete references to DoD standards and terminology remained in the core question set; instruction to the team included logical interpretation of these terms in today's lexicon. The team leader also discussed the criteria for assessing responses and substantiating evidence, and the goals and objectives of the SDCE. Finally, the evaluators were trained in the use of the hardware and software for creating and retaining their evaluations, which would be used to build consensus on the offerors' strengths, inadequacies, deficiencies, and risks.

To emphasize the Level 3-equivalence of the large SDCE, the training highlighted the Critical Capability Areas (CCAs) and Critical Capabilities (CCs) where core questions and criteria were documented. In addition, training covered questions and criteria for program-specific technologies and COTS software; these were selected or developed by the team leader and program office to elicit information on known risk areas.

5.3.1 Lessons Learned

Although the team leader provided extensive training on the SDCE, evaluation method, and overall assessment approach, the team noted several problems upon completing the evaluation. In general, they agreed that the team should have some refresher training before each step in the process. Since this SDCE lasted for over a year (from start of preparations to completion of the final report), and since some team members were part-time, some of the training, especially in consensus building, was forgotten or incompletely recalled. Even though all team members were members of the technical staff or government employees, some of the team members were not clear on the use of the computer system and intranet for reviewing the SDCE responses and documenting their findings. As a result, the team leader determined that team exercises, additional examples, hands-on training for the computer system, and refresher training should be part of the training plan for any future evaluations.

5.4 Evaluation Process

5.4.1 Teaming Arrangements

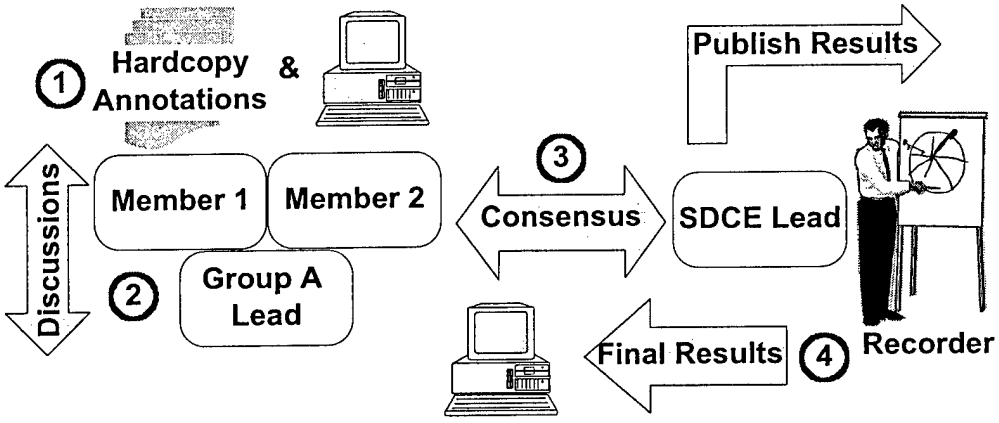
The first Level 3-equivalent SDCE done at SMC required a large team of evaluators and a significant amount of time to perform the evaluation of each criterion (see Subsection 5.4.4 *Resource Usage* under this section). To reach agreement on the outcome of the evaluations, the team leader developed a process that divided the SDCE into manageable portions for small sub-teams' evaluations, and then

implemented a consensus approach for the large team to finalize and record the evaluations, generate Evaluation Notices (Ens), and document strengths and/or issues in the final report.

Each question-criterion pair was evaluated by two evaluators on a sub-team, which yielded either a first level of agreement on the evaluation results or highlighted where engineering judgment differed on the criterion satisfaction. Individually, the evaluators were tasked to create a list of all criteria within their areas of responsibility that had not been met; generate their own observations, risks, strengths, and inadequacies; review all criterion/question pair write-ups by their sub-teammates; and summarize the observations, strengths, inadequacies, and risks. Each evaluator also noted the findings that seemed to be common to the CC, the complete set of CCs assigned to the sub-team, and the SDCE as a whole.

5.4.2 Consensus Building

After the individual and sub-team evaluations were completed, the entire evaluation team met to review the evaluations done within the sub-teams and reach consensus on the findings. The team leader moderated the discussion by CC, where the sub-team responsible for the CC presented their evaluations of the criteria, along with strengths, inadequacies, observations, risks, and issues relative to the offeror's response. All team members were encouraged to participate in these consensus meetings to discuss interrelationships within and across criteria that may have been evaluated by different sub-teams, and to reach agreement on the evaluation outcome. An important part of these meetings was achieving consensus on whether or not each criterion had been satisfied. During the consensus process, the team also identified topics where additional information was needed. These topics became the basis for ENs sent to the contractors for additional information. Figure 5-1 depicts the consensus process used for the large SDCE.



Consensus Meeting Preparation

1. Individual team members perform assigned evaluation and documentation

2. Individual team members perform intra- & inter-group discussions as necessary

Consensus Meeting

3. SDCE team lead moderates criteria discussion with entire SDCE team
Assigned recorder documents consensus points on flip chart(s)

4. SDCE team leaders convert flip chart annotations into final consensus for team review

Figure 5-1. SDCE team consensus process.

When responses to the ENs were received from the contractors, a similar, but shortened, process was used. All team members reviewed the entire set of EN responses since the ENs covered broad topics that crossed CCs, CCAs, and even FAs. Individual team members updated their documented strengths, weaknesses, and risks based on their review of the EN responses. Sub-teams then met to determine whether there was any change in satisfaction status for their criteria. The full team then met in another set of consensus meetings for the EN responses. During this meeting, consensus was achieved on the addition or deletion of strengths, weaknesses, and risks, and any changes in criterion satisfaction status based on the EN responses.

5.4.3 Roll-Up Process

One of the lessons learned from earlier SDCEs was integral to the accomplishment of the SDCE pilot; i.e., rolling up individual strengths, inadequacies, and weaknesses vertically to functional areas does not provide the evaluation team with insight into deficiencies that span the SDCE engineering, technical, and program management domains. As the team worked through the SDCE responses, they found that individual inadequacies and weaknesses were generally symptoms of larger underlying problems, or "issues," that cut across the functional area boundaries.

In the roll-up process for the large SDCE, the team synthesized the individual symptoms across the FAs to derive the larger issues that were documented in the final reports to the contractors. To facilitate this task, the team leader kept lists of individual strengths, inadequacies, observations, and risks during the consensus meetings. Once all criteria had been evaluated and the team's criteria evaluation documented, the team leader "bucketed" the symptoms into "issue bins" that related multiple symptoms to a single problem. After "bucketing" all team findings into an issue or issue category, the bucket contents were synthesized by the team leader, reviewed by the team members, and documented in the final reports to the offerors to facilitate their internal process improvement prior to the Engineering and Manufacturing Development (EMD) source selection.

The issue bin list for the large SDCE is provided in Table 5-1.

Table 5-1. Issue Bins and Categories

Issue Type	Categories
Process	Software team members and responsibilities
	Software item definition and management
	Integrated Product Team (IPT) structure and definition
	Process definition, especially across IPTs and team members
	Life cycle model
	Quantitative project management (e.g., cost, schedule, effort, metrics)
	People/group interface management
	Training
	Peer reviews
	Multi-site software development
	Quality assurance
	Configuration management
	Risk management
	Subcontractor management
Product Engineering	Requirements analysis and management
	Computer system architecture and design
	Testing approach and management (integration and verification)
	Interfaces
	Specialty engineering, especially RMA and supportability
	Traceability
	Operations and maintenance approach
	COTS and reuse software
	Open systems
	Distributed network-based systems
	Trusted systems
	Artificial intelligence
SDCE Response	N/A
Government	N/A

5.4.4 Resource Usage

The first large SDCE was completed outside of source selection (see Figure 3-6) primarily due to the schedule and resource constraints that were imposed by the size and complexity of the evaluation. As presented at the Software Technology Conference in May 2002, this SDCE required over 9,000 hours to complete.[21] The breakdown of hours and resources used is shown in Figure 5-2.

21 [Eslinger 2002] Eslinger, S., *Piloting the Level 3-Equivalent SDCE: The Evaluation Team Perspective*, Software Technology Conference 2002.

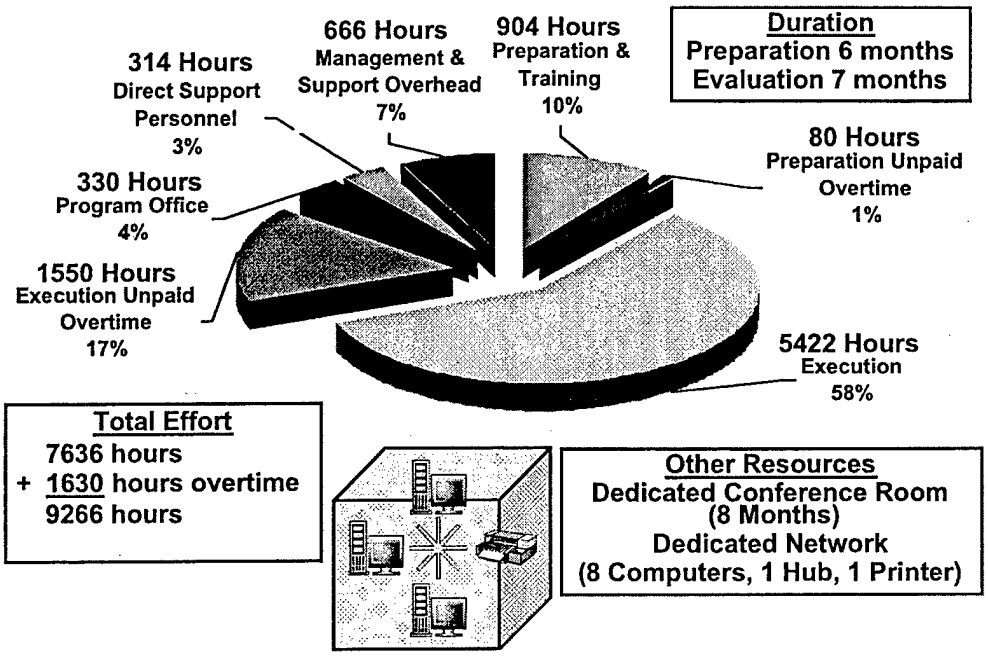


Figure 5-2. SDCE resources and duration (AF & Aerospace).

5.4.5 Lessons Learned

The resources used to perform the Level 3-equivalent SDCE left a severe shortfall in the ability to staff other competing programs with adequate acquisition support. Since this was the pilot for the Level 3-equivalent SDCE, the acquisition organization within the company decided that their most experienced staff should perform the evaluation—those who had other pressing requests from their management and programs. Rather than drop these requests, the experts provided over 1600 hours of unpaid overtime to complete the evaluations. Their findings concluded that: (1) even if the resources are available for the duration of the evaluation, the Level 3-equivalent SDCE requires an exorbitant amount time and effort, which far outweighs the benefits of the evaluation; and (2) a significantly smaller SDCE would have achieved the program goal of having the best software processes in place prior to the start of the next acquisition phase.

Appendix A—Basic SDCE

A.1. SDCE Cover Page

Annex <*X*> to Section L
<Insert Contract Number Here>

<Insert Program Name Here>

Software Development Capability Evaluation

<Insert RFP Release Date Here>

A.2 SDCE Instruction Text

SOFTWARE DEVELOPMENT CAPABILITY EVALUATION (SDCE)

In order to assure that offerors have the software development capabilities required for the ***<insert program name>*** program, the Government will conduct a Software Development Capability Evaluation. The SDCE will be conducted with the prime offeror and proposed software team members[22] who have significant software development responsibility.[23] This evaluation will be based on an analysis of the documentation described below that is to be submitted with the offerors' proposals. The collection of substantiating documents supplied shall include substantiating documents from the prime offeror and all significant software team members.[24] For instances of teaming and prime/subcontractor arrangements among offerors, it is the responsibility of the prime offeror to determine the required information (such as proposal information, SDCE question responses, and supporting data) that is to be supplied to the Government by each member of the bidding team.

The offeror shall submit an electronic media copy of the SDCE on CD-ROM. The offeror shall provide an original and N paper copies (each identified by copy number) of the SDCE. The offeror may submit both the paper copies and electronic versions of the SDCE in offeror format. If electronic versions of supporting data are not available, the offeror may submit paper copy only for that piece of data.

The following information in direct support of the SDCE is to be submitted with the proposal and will not be limited by the specified page counts for the proposal:

1.0 OVERVIEW OF THE *<insert program name>* SOFTWARE DEVELOPMENT EFFORT

The prime contractor shall provide an overview for the total ***<insert program name>*** software development effort that addresses: the organization of the contractor team, including SQA, process groups, etc.; the task and responsibility distribution among

---

[22] A *software team member* is any internal or external organization that develops, tests, or supports software-related work being performed for this contract and has an agreement (formal or informal) with the prime contractor or any subcontractor. These organizations include, but are not limited to, intra-corporation software organizations, in-house service providers, developers, fabrication/manufacturing organizations, laboratories, and subcontractors. Examples of an agreement include a contract, work authorization, memorandum of agreement, or oral agreement.

[23] *Significant software responsibility* includes responsibility for any deliverable software (including the software portion of firmware) or for any software used in satisfying, verifying or validating requirements or used in performing or supporting operations or sustainment (e.g., applications, security, safety, training, simulation, analysis, database support, automatic test equipment, maintenance).

[24] A *significant software team member* is a software team member with significant software responsibility.

the team members; and the processes used to manage and control team member performance. The purpose of this overview is to provide a foundation for review of the SDCE questionnaire responses. All information in this overview shall be consistent with information provided in the planning documentation and other volumes of the RFP response, and shall reference such information where appropriate. Specific volume, page, and paragraph numbers are required where references are used. The overview shall be limited to eight (8) pages.

2.0 RESPONSES TO QUESTIONS

Responses to the questions (see tailored questions in attachment 2) are encouraged to reference the documentation accompanying the proposal, such as the draft SDP or IMS, or other proposal volumes. This approach is intended to reduce the SDCE preparation effort and eliminate duplication within the proposal. When responses to the SDCE questions reference documentation accompanying the proposal, specific page number and paragraph references shall be provided with the response to the question.

Responses should be concise and unambiguous, not exceeding two pages per question. Responses should be provided for the processes to be employed on the ***<insert program name>*** program by the offeror and any significant software team members. Common processes among the team members are required to be described only once. For processes not common among the software team members, each non-common process shall be described in the response. The total page count for each question response with non-common processes shall not exceed three pages per question. In this case, the question response shall clearly indicate how the non-common processes would be effectively integrated across the team.

The response to one question may refer to the response to another, when appropriate. Each response should reference supporting data that define the process or provide evidence of implementation; this may be done in the question response or in a separate cross-reference matrix. The text of each question and criteria should precede the contractor's response to each question. The text will not count against the response page limits.

3.0 SUBSTANTIATING DOCUMENTS FOR EXISTING PROCESSES

Substantiating documents must be submitted for all existing processes planned for use on the ***<insert program name>***, whether employed by the prime offeror or software team members. This substantiating documentation is not page limited.

Examples of substantiating documents include:

- Copies of corporate software-related procedure, process, standard, and practice descriptions that are relevant to the acquisition. (Also for each

software team member if different procedures, processes, or practices are to be employed.)

- Copies of documents that provide evidence of use of the proposed processes (e.g., development schedules, software development plans, software requirements specifications, test and integration plans and procedures, peer review minutes, metrics reports).

Evidence of use shall be provided for each question/response from two (2) projects. Sample data documents provided as evidence of use of the proposed processes may be obtained from the current project (if work has begun) or from other projects.

4.0 DESCRIPTION OF NEW PROCESSES

For new processes not yet documented, describe the benefits and risks of using the new processes and the rationale for employing them in lieu of examples of past application. This description will not count against the SDCE response page limits.

5.0 DATA INDEX AND SAMPLE DATA COVER SHEET

The following forms must be completed and submitted with the proposal: (This data will not count against the SDCE page limits.)

- An index of all supporting material submitted along with a reference to where that supporting material can be found.
- Cover Sheet for Project Sample Data for each sample submitted (see attachment #1).
- Cross-references between the questions in the SDCE and the specific portions of the sample material submitted that answer the questions or provide evidence of their implementation. If specific references are not provided (by page number and/or paragraph number(s)), the referenced evidence will not be considered in the evaluation. References to evidence that must be accessed via the Internet or an intranet will not be considered in the evaluation.

The format and method of providing the index and cross-references is at the discretion of the contractor.

Attachments:
Cover Sheet for Project Sample Data (Attachment 1)
Tailored SDCE Questions (Attachment 2)

Attachment 1: Sample Data Cover Sheet

This attachment contains an example that illustrates how to complete the Sample Data Cover Sheet and one copy of a blank Sample Data Sheet.

EXAMPLE

Cover Sheet for Project's Sample Data		
Contractor: Team A		
Sample Project Name: Project X		
Sample Project Customer: U.S. Air Force Space and Missile Systems Center		
Critical Capability(ies): 4.4.2 Metrics Application		
Title of Sample: Project X Software Development Metrics Reports		
Explain why your experience on the sample project is relevant to the proposed project. Object-oriented methods and metrics were used on the sample project. The same object-oriented methods and metrics are planned for use on the proposed project.		
ATTRIBUTES	**PROPOSED PROJECT**	**SAMPLE PROJECT**
Application Domain	Weather Satellite	Communications Satellite
Product Type	Ground System (Command and Control)	Ground System (Command and Control)
Acquisition Phase[1]	EMD	EMD
Software Development Phase(s)	Design; Coding and Unit Test	Coding and Unit Test, Increments 1 and 2
Award Date		1/94
Contract Duration	8 Years	5 Years
Current Project Phase/ Contract Month[2]		EMD: Between System PDR and System CDR/Month 24
Prime/Subcontractors[3]	2 Software Subs	Prime & 1 Software Sub
Software KSLOC[4]	750	500
Language(s) and Percentages	Ada 95: 90%, C++: 10%	Ada 83: 75%, C++: 25 %
Target Processor(s)/OS(s)	RISC 6000/UNIX	VAX 6200/VMS 6.2
Applicable Standards	IEEE 1498	DoD-STD-2167A & 2168

[1]For "Proposed Project," phase(s) in which Critical Capability(ies) are to be used; for "Sample Project," phase in which sample was generated.

[2]Phase/month of the Sample Project as of the current date.

[3]Contractors developing the software products specified in the "Product Type" row

[4]Total number of KSLOC for software specified in the "Product Type" row

Cover Sheet for Project's Sample Data		
Contractor:		
Sample Project Name:		
Sample Project Customer:		
Critical Capability(ies):		
Title of Sample:		
Explain why your experience on the sample project is relevant to the proposed project.		
ATTRIBUTES	**PROPOSED PROJECT**	**SAMPLE PROJECT**
Application Domain		
Product Type		
Acquisition Phase[1]		
Software Development Phase(s)		
Award Date		
Contract Duration		
Current Project Phase/ Contract Month[2]		
Prime/Subcontractors[3]		
Software KSLOC[4]		
Language(s) and Percentages		
Target Processor(s)/OS(s)		
Applicable Standards		

[1]For "Proposed Project," phase(s) in which Critical Capability(ies) are to be used; for "Sample Project," phase in which sample was generated.

[2]Phase/month of the Sample Project as of the current date.

[3]Contractors developing the software products specified in the "Product Type" row

[4]Total number of KSLOC for software specified in the "Product Type" row

Attachment 2: SDCE Questionnaire

Question numbers are consistent with AFMCPAM 63-103. Revised questions/criteria are identified with an (a), i.e., Q3a instead of Q3 and C3a instead of C3.

1. Program Management

1.2 Program Planning and Tracking

1.2.1 Planning

Q1a How is your software development planning integrated with systems management and hardware management? **C1a C3**

Q3a Describe your technical and management reviews used to control the development progress throughout the entire development period. Define these events and corresponding criteria. How are these events incorporated into program documents and into which program documents are they incorporated? (e.g., IMP, IMS, SDP)? **C3 C4a**

Q5a Identify the software tracking metrics to be used on this program. Describe your process for monitoring and reporting critical status metrics or indicators. How do you determine when management action is required? Describe the conditions that would result in a management action for each established metric or indicator. **C6a**

C1a The program planning adequately accounts for the integration of software development and management with system and hardware management. **Q1a**

C3 The program planning includes the necessary reviews, accountability, status assessment, schedule control and reporting to manage the software related system development activities leading to the definition of the software requirements baseline. **Q1a Q3a**

C4a The program planning includes an adequate series of technical and management reviews with associated completion criteria (including quality gates) that are used to control the development progress. **Q3a**

C6a An effective metrics identification and monitoring process is documented, adequate metrics are in place, and variance thresholds are established for critical status metrics (e.g., size, defect detection, defect removal, effort, cost, progress, and schedule). **Q5a**

1.2.4 Schedules

Q1a Describe your approach to establishing the software development schedules from the top system level schedule to the lowest level detail schedules. **C1**

Q5a Describe your method for monitoring and statusing software development schedules. Who is responsible for this function? Which level of schedule that addresses software is used as the baseline to track and report status? **C1**

C1 Software schedules are established in sufficient detail to maintain visibility and control of the development process including the establishment of any planned blocks, builds or increments. **Q1a Q5a**

1.3 Subcontractor Management

1.3.2 Program Management, Subcontractor Development Management

Q1a Fully describe your process for subcontractor and/or COTS vendor management including reporting and control of the subcontractor and/or COTS vendor software development activities. How does this process relate to and integrate with your overall system program management approach? Describe how the subcontractor and/or COTS vendor management and review activities are reflected in the program level IMS. **C1a**

C1a The proposed subcontractor management process is integral to the system program management process and provides integrated reporting and control of the subcontractor software and/or COTS vendor development activities consistent with the program's management control system. **Q1a**

1.5 Risk Control

1.5.1 Risk Identification

Q1a Describe your process to identify, manage, and reduce risks associated with the system and software development. **C1a**

Q2 Identify the projected risks and short falls associated with this program as a result of applying this process. **C1a**

C1a An effective process is defined and is used to identify the short-falls and risks associated with the proposed development activities, and effective means are being employed to manage and mitigate the significant identified risks. **Q1a Q2**

---

2. Systems Engineering

2.1 System Requirements Development, Management and Control

2.1.3 Requirements Change Control

Q2 Describe the requirements change control process, with reference to both internally and externally generated changes. **C2**

Q3 What process is used to control allocation of changed (new or existing) requirements between hardware and software? **C3**

C2 All changes to requirements, including those generated by the customer, are managed by means of a defined change process. **Q2**

C3 Allocation of new and additional requirements between hardware and software is managed by a structured change process; reallocation of existing requirements between hardware and software is managed by a structured change process. **Q3**

2.1.4 Requirements Traceability

Q1a Describe the process used to provide two-way requirements-to-requirements, requirements-to-design, and requirements-to-verification traceability throughout the system

life cycle. At what point are requirements-to-requirements, requirements-to-design and requirements-to-verification traceability established and documented? What provisions exist to maintain the traceability? **C1a**

C1a Two-way requirements-to-requirements, requirements-to-design, and requirements-to-verification traceability are effectively maintained from system specifications to hardware and software configuration item specifications, from specifications to design documentation, and from specifications to verification planning execution, and the information is effectively shared and used. **Q1a**

2.5 Systems Engineering Planning

2.5.1 Methodology and Standards

Q1a Describe how the program's engineering policies, practices, procedures, and standards are defined, documented, and enforced; and how they relate to the corresponding software systems engineering policies, practices, procedures and standards. **C1a**

C1a Effective systems engineering policies, practices, procedures, and standards are defined, enforced, and are consistent with systems engineering contractual standards. Effective policy, practice, procedures, and standards integration exists among the systems engineering and software systems engineering organizations. **Q1a**

2.6 System Integration and Test

2.6.1 Integration and Test Planning

Q2 If system builds are planned, describe how test planning for each system build includes the multiple levels of system integration and test (from units to CSCIs to subsystem to system-level test). **C2**

C2 Test planning for each system build includes the multiple levels of system integration and test (from units to CSCIs to subsystem to system-level test). **Q2**

2.7 Reuse

2.7.4a COTS/Reuse Software Evaluation, Selection and Management

Q1a Describe your process for evaluating and selecting COTS and reuse software, including the criteria that each product must meet before it is considered for inclusion in a development effort. **C1a C2a**

Q2a What is your approach for managing COTS and reuse software on this program? **C2a C3a C4a**

Q3a Describe how your software configuration management plan includes the configuration control of COTS and reuse software products selected for use on this program. **C5a**

C1a The contractor has a well defined process for COTS and reuse software selection that includes effective criteria to ensure that the selected products provide needed capabilities and meet system and software constraints within an acceptable level of risk. **Q1a**

C2a The contractor has appropriately considered the system life-cycle costs in the evaluation, selection and management of COTS and reuse software. **Q1a Q2a**

C3a The contractor has an effective plan for managing COTS and reuse software that is appropriately integrated with the software development plan and systems engineering management plan. Well-defined software processes have been suitably adapted to include COTS- and reuse-specific processes, standards, and procedures. **Q2a**

C4a The COTS and reuse software management plan adequately covers planning for systems engineering considerations, such as supportability, security, safety, and fault detection and isolation. **Q2a**

C5a The contractor's software configuration management plan adequately incorporates processes for installing COTS and reuse software on multiple hardware platforms, managing the configuration of multiple baselines, and controlling the licensing of COTS and reuse software products. **Q3a**

---

3. Software Engineering

3.3 Software Requirements Management

3.3.1 Software Requirements Analysis

Q1a Describe the software requirements analysis process(es) to be applied. Identify the specific methodologies and tools to be used to support the analysis process. What organizational element is responsible to perform the analysis? Identify the input to and output product from the analysis. **C3a**

C3a The selected requirements analysis methodology/methodologies is/are appropriate for the development effort, and compatible with other methodologies applied on the program. The analysis methodology is supported with necessary tools. **Q1a**

3.3.2 Software Requirements Changes

Q1 Describe the software development activities that result from a change in or addition to the requirements. When do they get performed? How do you ensure that they are performed? **C1a**

C1a The software development artifacts' (e.g., requirements, design, code, documentation) are appropriately revised as changes to the requirements are incorporated. **Q1**

3.4 Software Design

3.4.1 Design Methodology

Q1a Describe the process(es) and specific methodologies used to develop the software design. Describe how the methodologies interact with the requirements process, are used to maintain the design through development and are used for life cycle support. What tools are used to support the methodologies? **C1a**

Q3a What mechanism and format are used to describe the static structure and dynamic behavior of the software (e.g., execution priorities of the different components, and the execution control)? **C2a**

C1a Effective methodologies are used to develop, document and maintain the software design and interface with requirements processes. The methodologies are effectively supported by tools. **Q1a**

C2a The design description effectively incorporates the static structure and the dynamic behavior of the software. **Q3a**

3.5 Software Coding and Unit Testing

3.5.1 Code Development

Q7 What processes and procedures are used to ensure that the design is implemented completely and correctly? At what component level? Who has that responsibility? **C3**

C3 The developed software is unit tested. Realistic resources and schedules are allocated to this level of testing. Units are tested in all increments of development. **Q7**

3.5.2 Code Changes

Q2a Describe your process for estimating the effect of code changes on other parts of the system. What tools are used? Who is involved in the process? **C2a**

C2a Code changes are effectively reviewed for correctness and to avoid undesired impact on other software and system components. **Q2a**

3.6 Software Integration and Test

3.6.1 Software Integration

Q1a Describe your process for planning the software integration. How do you determine the order for integrating the different software components? Describe how your integration process accommodate all levels of software integration, how integration changes are handled and how software integration processes support hardware/software integration. **C1 C2 C4a**

C1 The software integration planning takes into account the interdependencies between the different software components and the criticality of each component. **Q1a**

C2 The software integration planning takes into account the availability of other components of the system. **Q1a**

C4a The software integration planning and process effectively accommodate software integration at all levels, effectively incorporate integration changes and support hardware/software integration. **Q1a**

3.6.2 Software Testing

Q1a How are verification plans, verification procedures, and verification cases developed? When? By whom? Where are they documented? How are they reviewed? How are they controlled? Verification includes all verification methods (i.e., inspection, analysis, demonstration and test). The answer to this question should include all applicable verification methods. **C1a**

Q2a What tools will be used for verification? When will they be available? Will they require any special inputs? Will their outputs require any special processing? What is your process to ensure that all required verification resources have been planned and allocated as well as qualified for use? **C2a**

C1a The software verification process(es) adequately incorporate all applicable verification methods. **Q1a**

C2a A process exists to ensure that software verification is adequately planned with sufficient verification resources and that those verification resources are adequately qualified for their intended use. **Q2a**

---

4. Quality Management and Product Control

4.1 Software Quality Management

4.1.3 Software Discrepancies

Q2 Identify and describe specific procedures to identify, document, report, track, and resolve software discrepancies. **C1a**

Q3 Describe your method for resolving software versus hardware discrepancies in your problem reporting systems. **C1a**

C1a Effective, documented procedures exist to resolve software versus hardware discrepancies and to identify, document, track, and resolve software discrepancies. **Q2 Q3**

4.2 Software Quality Assurance (SQA)

4.2.1 SQA Organizational Approach

Q1 Describe the responsibilities of the SQA organization and how it interfaces with other organizations. **C1 C2a**

Q5 What mechanisms and channels exist for SQA to surface quality problems and elevate them in the management chain until they are resolved? **C3**

C1a An SQA organization is assigned the responsibility to monitor the software development process and the software products. **Q1**

C2a The responsibilities, mission, and interface(s) of SQA with program management, engineering, configuration management, and test functions are adequately defined and documented. **Q1**

C3 The SQA group is empowered to effect changes to the program when quality goals are not followed. **Q5**

4.2.2 SQA Staffing

Q1a How many SQA personnel are normally assigned to a major program? **C1a**

C1a An adequate number of technically knowledgeable and trained SQA personnel are staffed to the program to accomplish assigned responsibilities and functions as proposed for this program. **Q1a**

4.2.3 Compliance Checking

Q2a Describe how SQA ensures compliance of software development activities with defined processes and how discrepancies are resolved. Which processes are audited? How often? **C2a**

Q4 Describe how SQA verifies that the software products adhere to the program's requirements, standards, and quality goals. **C3**

C2a Adherence to defined software development and management processes is verified and discrepancies are monitored until corrected. **Q2a**

C3 SQA audits designated software work products to verify compliance with quality goals and adherence to the applicable standards and requirements. **Q4**

4.3 Defect Control

4.3.1 Defect Activity Coordination

Q1 Describe your program plan for preventing software defects. **C1a**

C1a The program develops, applies and maintains a plan for its defect prevention activities. **Q1**

4.3.2 Defect Collection and Analysis

Q1 Describe your approach to collection and analysis of defects. **C1 C2**

Q5 Identify your approach to collecting defects resulting from peer reviews, testing, and design reviews. Is this approach contained in the quality plan? **C3**

C1 Common causes of defects are identified, prioritized, and systematically eliminated. **Q1**

C2 Causal analysis meetings are conducted. **Q1**

C3 Data on defects identified in peer reviews, document review, and testing are collected and analyzed. **Q5**

4.5 Peer Reviews

4.5.1 Peer Review Planning

Q1a Describe the documented internal peer review procedures and requirements, including products that require peer reviews, definition of required participants, completion criteria and review content and follow-on action item resolution. **C1a**

C1a Internal documents exist that: identify appropriate required participants in the reviews, provide adequate specific criteria for successful completion, describe adequate documentation required for the review and describe how follow-on actions are adequately documented, tracked and controlled. **Q1a**

4.7 Software Configuration Management (SCM)

4.7.2 Baseline/Configuration Identification and Management

Q1 How are software baselines, both formal and informal, controlled using documented procedures for software and documentation and for transfer to other libraries, where appropriate? **C1a**

Q5a What is the program approach to establishing and controlling formal and informal developmental baselines and verification configurations? **C4a**

C1a The configuration control implementation establishes a developmental configuration for each software product under development or maintenance, effectively controls the preparation and dissemination for changes to the master copies of deliverable software and documentation, and maintains current copies of deliverable documentation and code. **Q1**

C4a Effective procedures exist and are followed to create and maintain formal and informal developmental and verification baselines. **Q5a**

4.8 Documentation

4.8.2 Technical Adequacy

Q2 What standards do you use in documenting test requirements? **C2a**

C2a Adequate standards exist for documenting test requirements for the software. **Q2**

---

5. Organizational Resources and Program Support

5.2 Facilities

5.2.1 Development Facilities

Q1 Describe the software development facilities (host development computers, workstations, networks, memory systems, etc.) intended for the program in terms of quantity, location, availability date, capacity and response time. Describe the level of integration of the system/software development facilities (environments). **C1a**

C1a An effective plan for establishing and maintaining the required system and software development facilities is documented, and is consistent with the program's requirements, needs, usage estimates, and schedule. **Q1**

5.6 Organizational Process Management

5.6.1 Process Planning and Coordination

Q3a Describe the coordination of the system development and software development process management activities of the organization and the responsible individuals or groups. How are these activities coordinated with the program? How is software process compliance enforced? **C2a**

C2a The system and software process management activities of the organization are effectively coordinated and enforced; in particular these activities include:

- Defining and managing changes to the organization's system and software processes;
- Collecting and maintaining data on use of the organization's system and software processes;
- Direct feedback to management on the program's software process activities to ensure compliance and effective use. **Q3a**

5.7 System/Software Engineering Environment

5.7.2 S/SEE Components

Q3a Describe how each tool in the S/SEE supports the software development process functions and methodologies selected for the program and the relationship of the S/SEE to the life cycle maintenance environment. **C1a**

C1a The S/SEE components effectively support the program's software engineering development and management requirements, functions, methodologies, and activities, and will effectively support the maintenance environment when operational. **Q3a**

Appendix B—Instructions for the Large SDCE

B.1. SDCE Cover Page

Attachment 1: SDCE Instructions

<Program Name>

Software Development Capability Evaluation (SDCE)

SDCE Release Date
<MM/DD/YYYY>

B.2. SDCE Instruction Text

1. Introduction

In order to assure that the contractors have the software development capabilities required for the *<Program Name>* program, the Government will conduct a Software Development Capability Evaluation (SDCE). The SDCE will be conducted with the prime contractor and software team members[25] who have significant software responsibility.[26] The SDCE will apply to the following categories of software: onboard software (e.g., spacecraft, communications, payload); ground mission software (e.g., mission planning; mission data processing; event validation and reporting; telemetry, tracking and commanding); and other software used in satisfying, verifying, or validating requirements or used in performing or supporting operations or sustainment (e.g., training, simulation, analysis, database support, automatic test equipment, maintenance). This evaluation will be based on an analysis of the questionnaire responses and the substantiating information that is submitted as part of the prime contractor's SDCE package. For instances of teaming arrangements among contractors, it is the responsibility of the prime contractor to determine the required information (such as overview information, SDCE question responses, substantiating information, and supporting data) that is to be supplied to the Government by each member of the team. The collection of substantiating information supplied shall include substantiating information from the prime contractor and all significant software team members.[27] The SDCE responses shall cover processes to be used over the entire *<Program Name>* development life cycle, including the Program Definition/Risk Reduction (PD/RR) and Engineering and Manufacturing Development (EMD) phases.

2. Instructions for Responding to the SDCE

The contractor shall supply information as described below.

---

[25] A *software team member* is any internal or external organization that develops, tests or supports software-related work being performed for this contract and has an agreement (formal or informal) with the prime contractor or any subcontractor. These organizations include, but are not limited to, intra-corporation software organizations, in-house service providers, developers, fabrication/manufacturing organizations, laboratories, and subcontractors. Examples of an agreement include a contract, work authorization, memorandum of agreement, or oral agreement.

[26] *Significant software responsibility* includes responsibility for any deliverable software (including the software portion of firmware) or for any software used in satisfying, verifying or validating requirements or used in performing or supporting operations or sustainment (e.g., applications, security, safety, training, simulation, analysis, database support, automatic test equipment, maintenance).

[27] A *significant software team member* is a software team member with significant software responsibility.

2.1 Overview of the <*Program Name*> Software Development Effort

The prime contractor shall provide an overview for the total <*Program Name*> software development effort that addresses: (a) the organization of the contractor team, including all software team members, and organizational support groups (e.g., SQA, process groups); (b) the task and responsibility distribution among the software team members; and (c) the processes used to manage and control team performance. The overview should address the total <*Program Name*> development life cycle. The purpose of this overview is to provide a foundation for review of the SDCE questionnaire responses. All information in this overview shall be consistent with information provided in the Integrated Master Plan(s) (IMP(s)), Software Development Plan(s) (SDP(s)), and other material provided as part of the SDCE package, and shall reference such information where appropriate. Specific volume, tab, page, and paragraph numbers are required where references are used. The overview shall be limited to eight (8) pages.

2.2 SDCE Questionnaire Responses

2.2.1 General Information

The SDCE questions and criteria are listed in Attachment 2, in order by one-digit Functional Area (FA), two-digit Critical Capability Area (CCA), and three-digit Critical Capability (CC). Each question is followed by a list of criteria within that CC that could be used for that question. Each criterion is followed by a list of questions within that CC that could be used with that criterion. If a question or criterion is not explicitly contained in Attachment 2, it will not be used in this SDCE. For example, 1.3.2 Question 1 lists C1 and C3. C1 is not contained in the list. It will not be used in this SDCE. No source other than Attachment 2 is needed for the questions and criteria for this SDCE.

A core set of 130 questions and 118 criteria has been determined to be equivalent to Level 3 of the Software Engineering Institute's Capability Maturity Model for Software (SW-CMM). The questions and criteria in this core set are designated by "[CORE]" in front of the text of the question or criteria. Questions indicated by "[CORE]" are mandated by the Deputy Undersecretary of Defense for Acquisition and Technology DUSD (A&T) for SW-CMM Level 3-Equivalence in the May 22, 2000 memorandum.

The questions have been divided into three (3) categories for convenience: institutionalization, basic, and project-specific technology. The *institutionalization* questions are the SW-CMM Level 3-equivalent core set questions (in Functional Area 5) that are denoted with "[Inst]" and "[CORE]" in front of the text of the question. The *basic* questions are the rest of the questions listed in Functional Areas 1–5. Basic questions include both core and non-core questions. The *project-specific technology* questions are those listed in Functional Area 6.

The text of each question (including any CC preamble shown in Attachment 2 within square brackets, e.g., 5.1.1) and its associated criteria should precede the SDCE response. Answer all parts of the question and criteria. Responses to the questions are encouraged to be provided directly in the substantiating information accompanying the SDCE response, such as in the contractor's SDP or other evidence. This approach is intended to reduce the SDCE preparation effort and eliminate duplication within the SDCE package. When responses to the SDCE questions are provided in the documentation accompanying the responses, specific volume, tab, page number, and paragraph references shall be provided with the response to the question. Responses should be concise and unambiguous. Responses shall be provided for the processes to be employed on the <*Program Name*> program by the prime contractor and all significant software team members. Common processes between the prime contractor and significant software team members require only one response. However, for processes not common among the prime contractor and significant software team members, the combined responses shall not exceed the specified page limits. The response to one question may refer to the response of another, when appropriate. Each question response should reference, by volume, tab, page number, and paragraph reference, (1) substantiating information that defines the process and (2) substantiating information that provides evidence of use. These references to substantiating information shall be provided in the question response (for a small number of references) or in a separate cross-reference matrix (when there are many references).

The prime contractor and each significant software team member shall answer the 23 institutionalization core questions. The responses are to be collected into one complete response for each of these questions. Each portion of the response shall be labeled to identify the organization to which it applies (e.g., ABC Company, XYZ Division). The prime contractor and each significant software team member shall provide the corporate/organization substantiating information for the institutionalization questions. The prime contractor and each significant software team member shall each provide a bi-directional cross-reference of that corporate/organization information linked to their institutionalization question responses.

The questions in Functional Area 6 are on four (4) program-specific technologies. The prime contractor shall select those technology areas that apply to their <*Program Name*> approach for the entire life cycle. For each technology area selected, all 13 questions shall be answered. The responses for the technology areas chosen shall encompass the prime contractor and all significant software team members to which the technology applies.

In the questions and criteria:

1. All references to "SEMP," "SEMS," "SEDS" shall be interpreted as "IMP," "IMS," and "detailed schedules," respectively.
2. All references to "CSCI" shall be interpreted as "software item."
3. All references to "subcontractors" shall be interpreted as "software team members" as defined in the footnote on page 1 of these instructions.

2.2.2 Response Page Limits and Organization

The following instructions shall be observed in preparing responses.

- Table 2, Response Submissions, specifies the page limits for overview and question responses. The institutionalization question responses shall include a combined response from the prime contractor and all significant software team members.

- The prime contractor determines the allocation of pages for any given question response. Note that substantiating information is not subject to this page limit.

- All references to substantiating information must be unique and unambiguously identify the applicable substantiating item and the information contained in the substantiating item that is to be used for evaluation. In particular, the specific location (e.g., volume, tab within the volume, page, and paragraph numbers) shall be supplied in the reference. If extraneous information unrelated to substantiating SDCE questions is present in the reference material, it shall be unambiguously identified as such, either as part of the specific reference to the substantiating information, or in the material itself. For example, if an extracted portion of a document is provided, the reference should point to only the relevant parts, or unrelated material should be noted as such in the extracted material. Enough of the referenced document shall be provided to provide context for the extracted portion. References shall only be made to substantiating information provided in the SDCE package.

2.2.3 Response Formatting

The following formatting shall be observed in preparing the hardcopy for the overview and SDCE responses:

1. Responses shall be typed single spaced without columns using black Arial font. The font size used shall be no smaller than 12 pt in height. Margins on each page shall be 1 inch on all sides. Kerning modification or other

techniques to reduce character size or spacing are prohibited. All text within illustrations and tables shall be Arial and at least 10 pt in height. Figure and table titles shall be at least 10 pt in height.
2. No foldouts are allowed for question responses.
3. Page limits are based on 8.5 x 11 inch paper with page setup at 100%.
4. No sound or video files may be used. Use of scanned images shall be minimized and embedded graphics shall be kept as simple as possible.
5. Responses shall be printed on one side only and shall be bound loose leaf.
6. The responses shall be organized in the same order in which the questions appear in Attachment 2.
7. There shall be one response for each question.
8. The question responses shall have consecutive page numbers.

The following formatting shall be observed in preparing electronic SDCE responses:

1. The electronic copies shall be provided in the Adobe Acrobat Portable Document Format (*.pdf) file format. These copies shall be delivered on 5-inch CD-ROM media. There shall be two (2) copies of the CD-ROM(s).
2. No sound or video files may be used. Use of scanned images shall be minimized and embedded graphics shall be kept as simple as possible.
3. Electronic responses and the file groupings (if any) shall be organized in a fashion parallel to the hardcopy response. Use of hyperlinks and bookmarks is encouraged but not required. If used, they shall be structured to enhance the ability of the reviewers to locate references and content intended for review.
4. References made to information contained in different SDCE responses shall be structured to work identically whether a reviewer obtains the referenced information from the hardcopy or electronic versions of the delivered material. The following example assumes that the material is available in both hardcopy and electronic format. If a reviewer is using the delivered hardcopy and comes to a reference to substantiating information elsewhere in the delivered SDCE material, then using only that specific reference, the reviewer should be able to find the exact same referenced information in the hardcopy version of the SDCE material or the electronic version of the SDCE material. This property of the reference must also work this way if the reviewer is using the delivered electronic version and needs to look up the reference in the hardcopy.

2.3 Substantiating Information

2.3.1 Types of Information

Substantiating information is intended to demonstrate institutionalization and effectiveness of proposed processes. This information shall cover all planned

processes, whether employed by the prime contractor or by the software team members. Samples of existing processes shall be directly relevant to the <*Program Name*> program's needs, relate to an SDCE critical capability, and demonstrate effectiveness in use.

Substantiating information includes:

- Corporate and organization information

  Copies of the corporate and organization software-related policy, procedure, process, standard, and practice descriptions that are being used on and relevant to this acquisition. If different procedures, processes, or practices are to be employed by various software team members, these shall be supplied as well.

- Evidence of Use on the Current <*Program Name*> PD/RR Contract and Two (2) Other Current or Past Contracts

  Copies of information that provide evidence of use of the proposed processes.

Existing processes. Substantiating documents must be submitted for all existing processes that are planned for use on the <*Program Name*> software development effort, whether employed by the prime contractor or by software team members. Results of previous SDCEs or SCEs will not be considered. However, the substantiating information used in a previous SDCE or SCE may be provided as substantiating information in this SDCE.

Evidence of use. Evidence of use on the current <*Program Name*> PD/RR contract shall be provided whenever it exists. In addition, evidence of use of proposed processes for other current or past contracts shall be provided from exactly two (2) contracts per question. Different contracts from different team members may be used on different questions. Table 1, Required Substantiating Information specifies the required substantiating information.

For the institutionalization questions, the substantiating process information for the organization shall be supplied from the prime contractor and from each significant software team member whether or not they will be used on this program. If the organization process information is the same as the corporate information or is tailored from the corporate information by reference, provide the corporate information. For the institutionalization questions only, do not provide substantiating evidence of use information from <*Program Name*> or other projects.

Description of New or Modified Processes. For new or modified processes not yet used, provide the process description. Then describe the benefits and risks of

Table 1: Required Substantiating Information

	Substantiating Information			
		Evidence of Process Use on Projects		
Question Type	Process Description(s)	*<Program Name>* PD/RR	Other Currrent/ Past Project	Another Current/ Past Project
Basic	X	X*	X	X
Institutionalization	X (Organizational)			
Project-specific technology	X	X*	X	X

* When it exists

using the new or modified processes and the rationale for employing them in lieu of examples of past application. This description will not count against the SDCE package page limits.

2.3.2 Classification

While unclassified substantiating information is preferred, substantiating information up to and including Secret collateral will be accepted if necessary. Any material classified above unclassified shall be submitted as one hardcopy only and no electronic copies. Substantiating information classified above Secret collateral shall not be submitted.

2.3.3 Substantiating Information Formatting

The following formatting shall be observed in organizing the hardcopy of the information used to support the SDCE responses and their evaluation.

1. For each Critical Capability Area, a cross-reference between each question and its associated substantiating information shall be submitted using the Capability Definition Matrix format. The format is shown in Attachment 1, Figure 3 and its accompanying instructions are shown in Attachment 1, Figure 4.

2. Hardcopies of substantiating information may be submitted in the contractor's format. Each project sample item submitted shall have a cover sheet attached in the format shown in Attachment 1, Figure 1.
3. Substantiating information shall be organized and presented in a manner that supports the SDCE evaluation.
4. The substantiating information hardcopies shall be bound in reasonable-sized (binders not larger than four (4) inches thick) volumes. These volumes shall be clearly identified as to their content on the cover and spine. The volumes and items within each volume shall be organized to facilitate the SDCE evaluation. Multiple copies of substantiating information are permitted if such redundancy makes the evaluation more efficient. If item redundancy is used, each instance of the item shall be uniquely identified.
5. Each volume shall have a table of contents at the beginning of the volume that covers the items it contains. The table of contents shall use the same format as the Sample Data Inventory (See Attachment 1, Figure 6).
6. Each individual item of substantiating information shall be uniquely identified and numbered with a scheme that is used consistently when the item is referenced by any SDCE response.
7. Each item of substantiating information shall be provided with a tabbed divider that contains the unique identification used for referencing that item. This divider shall be visible when the volume's covers are closed.
8. No pen and ink changes are allowed.
9. All items of substantiating information shall have page numbers.

A complete inventory of each item of substantiating information shall be provided using the Sample Data Inventory (SDI) format. The format is shown in Attachment 1, Figure 6 and its accompanying instructions are shown in Attachment 1, Figure 7.

The following formatting shall be observed in organizing the electronic copy of the information used to support the SDCE responses and their evaluation.

1. Electronic copies of substantiating information from the *<Program Name>* PD/RR contract are required. Electronic copy is to be submitted for non-*<Program Name>* PD/RR information only if the original information currently exists in electronic form or is available to the contractor in a format that can be readily converted into the required electronic standard. If electronic copies are not available, provide three (3) hardcopies as described in Sections 2.2.3 and 2.3.3 above for hardcopy material, except that substantiating information can be double-sided.

2. The electronic copy shall be provided in the Adobe Acrobat Portable Document Format (*.PDF) file format as described in Section 2.2.3 above.

Sample Data Inventory and Project Sample Data Cover Sheet. The following forms must be completed and submitted with the SDCE package: (This data will not count against the SDCE package page limits.)

1. A sample data inventory of all substantiating information submitted along with a reference to where that substantiating information can be found.
2. Cover Sheet for Project's Sample Data for each sample submitted (see Attachment 1, Figure 1).
3. Bi-directional cross-references between the SDCE questions and the specific portions of the substantiating information submitted that answer the questions or that provide evidence of their use. If specific references are not provided (by page number and/or paragraph number(s)), the referenced evidence will not be considered in the evaluation. References to evidence that must be accessed via the Internet or an intranet will not be considered in the evaluation.

The format and method of providing the cross-references is at the discretion of the contractor.

2.4 SDCE Package Delivery Information

2.4.1 General Information

SDCE packages will be read and evaluated using electronic and hardcopy information. To enable the Government to successfully view the packages electronically, the prime contractor shall submit the package files in the Adobe Acrobat Portable Document Format (PDF). Adobe Acrobat will be used to view the PDF files. The contractor has the option of generating "bookmarks" with each PDF file as well. The contractor shall provide hypertext links in a table of contents linked to each file provided in the SDCE package. Additional hypertext links within the SDCE package are encouraged for evaluating the responses. The use of bookmarks or additional hypertext links will not influence the evaluation. The electronic copies and paper copies are to be delivered to the address shown in the cover letter accompanying this attachment.

2.4.2 Required Items

The prime contractor shall submit both the paper copies and electronic versions of the SDCE materials in the format defined in these instructions, except for substantiating information that is only available in paper copy. If electronic versions of substantiating information are not available, the contractor shall submit only paper copies for that piece of data.

The following items are required as files on CD-ROM:

a. Overview

b. Responses to SDCE questions and criteria
c. *<Program Name>* Software Development Plan (SDP) for the prime contractor and each significant software team member
d. *<Program Name>* PD/RR Integrated Master Plan (IMP)
e. Bi-directional cross-reference of responses and substantiating information
f. Substantiating information, each project sample data item with a completed cover sheet (Attachment 1, Figure 1)
g. Completed cover sheet (Attachment 1, Figure 1) for each item of substantiating information unavailable in an electronic version (cover sheet for item hh below). These cover sheets must be organized identically with the hard copy.
h. Capability Definition Matrices
i. Sample Data Inventory

The following items are required as paper copy:

aa. Signed certification of virus checking (See Section 2.4.5)
bb. Overview
cc. Responses to SDCE questions and criteria (original and numbered copies)
dd. *<Program Name>* Software Development Plan (SDP) from the prime contractor and from each significant software team member (original and numbered copies)
ee. *<Program Name>* PD/RR Integrated Master Plan (IMP) (original and numbered copies)
ff. Bi-directional cross-reference of responses and substantiating information (original and numbered copies)
gg. Substantiating information provided in an electronic version (each project sample data item with a completed cover sheet)
hh. Substantiating information **unavailable** in an electronic version (each project sample data item with a completed cover sheet)
ii. Capability Definition Matrices
jj. Sample Data Inventory

2.4.3 SDCE Package Page Limitations and Number of Copies

The number of copies to be submitted for each item of documentation and the maximum page limitations are specified in Table 2. Page limitations will be strictly enforced. In the event a contractor exceeds the specified limit, the Government will not evaluate any pages in excess of the maximum. The page limitations apply to the Software Development Effort Overview and SDCE Question Responses. They do not apply to tables of contents, cross-reference information, Capability Definition Matrices, the Sample Data Inventory, or the Substantiating Information supplied with the responses.

Table 2 Response Submissions

Item	Number of Copies Required	Maximum Pages
Virus certification	1 hardcopy	
Overview of the Software Development Effort	7 hardcopies and 2 electronic copies	8 pages
SDCE Basic Question Responses	7 hardcopies and 2 electronic copies	290 pages
SDCE Institutionalization Question Responses	7 hardcopies and 2 electronic copies of the combined responses from the prime contractor and all significant software team members	80 pages
SDCE Project-specific Technology Question Responses	7 hardcopies and 2 electronic copies	30 pages/ technology area selected
<Program Name> SDP	7 hardcopies and 2 electronic copies	
<Program Name> PD/RR IMP	7 hardcopies and 2 electronic copies	
Substantiating Information - process descriptions for program - *<Program Name>* PD/RR - other projects - organization process descriptions	3 hardcopies* and 2 electronic copies (Electronic copy is to be submitted for all *<Program Name>* PD/RR information. Electronic copy is to be submitted for non-*<Program Name>* PD/RR information only if the original information currently exists in electronic form or is available to the contractor in a format that can be readily converted into the required electronic standard.)	
Bi-directional cross reference	7 hardcopies and 2 electronic copies	
Capability Definition Matrices	7 hardcopies and 2 electronic copies	
Sample Data Inventory	7 hardcopies and 2 electronic copies	

* See Subsection 2.3.2.

2.4.4 Paper Copies

The prime contractor shall provide an original and six (6) paper copies (each identified by Copy Number) of its Overview, Questionnaire Responses, Capability Definition Matrices, and Sample Data Inventory. The prime contractor shall provide

three (3) hardcopies of substantiating information (each identified by Copy Number), except as specified in Section 2.3.2.

2.4.5 Electronic Media

The prime contractor shall submit two (2) electronic media copies (original and one backup) of its SDCE package on 5-inch CD-ROMs. SDCE packages will be read and evaluated using the Adobe Exchange program operating under the Microsoft Windows 95 operating system. The prime contractor shall provide a signed certification that all files and electronic media have been checked for and are free of viruses. The prime contractor shall reference the anti-virus program name, version number, and date of data files. Each CD-ROM shall be properly externally labeled with disk name, brief description, copy number, and a range of the volumes of the paper copy covered on the CD-ROM (e.g., Vols. 11-20).

2.4.6 Evaluation Notice (EN) Page Format Restrictions and Limitations

These page formation restrictions shall apply to Evaluation Notices (EN):

1. The page limits for EN responses will be identified in the letters forwarding the ENs to the contractors.
2. Each page shall be counted with the exception of the transmittal letter, cover pages, blank pages, title pages, tables of contents, lists of figures and/or tables, or acronym lists.
3. The EN responses shall conform to the formatting specified in Section 2.2.3 above.

2.5 Site Visits

2.5.1 General Information

Site visits will be conducted for this SDCE. Site visits are performed after the SDCE team completes the SDCE response review. The SDCE team establishes the site visit schedule and agenda for each site visit. The discussion topics and ENs to be reviewed will be sent with the agenda to the respective prime contractor under a cover letter approximately three (3) weeks before the site visit. The site visit discussions are strictly limited to the topics submitted by the government. The prime contractor selects a single location for the site visit, depending on teaming arrangements. The site visit can be performed at the prime contractor's software development facilities or at a significant software team member's software development facilities. The prime contractor and each significant software team member are required to participate in the discussions.

It is the responsibility of the prime contractor to

1. Arrange meeting rooms, equipment, telephones, etc. for the government team
2. Ensure availability of requested documentation
3. Notify the host security office of impending visit
4. Ensure that security clearance information is sent to the government team
5. Ensure that clearances for the government team have been received
6. Assemble appropriate discussion participants from the prime contractor and each significant software team member
7. Prepare presentation materials
8. Review agenda with the government team
9. Ensure responses to ENs are provided
10. Ensure participation in program-focused discussions by the prime contractor and each significant software team member
11. Respond to the SDCE team feedback presentation

Attachments:

Forms

Project Sample Data Cover Sheet (Figure 1)
Blank Project Sample Data Cover Sheet (Figure 2)
Sample Capability Definition Matrix (Figure 3)
Capability Definition Matrix Instructions (Figure 4)
Blank Capability Definition Matrix (Figure 5)
Sample Data Inventory (Figure 6)
Sample Data Inventory Instructions (Figure 7)
Blank Sample Data Inventory (Figure 8)

Tailored SDCE Questions (Attachment 2) [See Appendix C]

This example illustrates how to complete the Project Sample Data Cover Sheet. This form must be provided with each piece of project sample data provided as substantiating evidence.

Cover Sheet for Project's Sample Data		
Contractor: Team A, ABC Company, XYZ Division, Rolling Hills, VT		
Sample Project Name: Project X		
Sample Project Customer[1]: U.S. Air Force Space and Missile Systems Center		
Critical Capability(ies): 4.4.2 Metrics Application		
Title of Sample: V4-4, Project X Software Development Metrics Reports		
Explain why your experience on the sample project is relevant to the *<Program Name>* project. Object-oriented methods and metrics were used on the sample project. The same object-oriented methods and metrics are planned for use on the *<Program Name>* project.		
ATTRIBUTES	***<PROGRAM NAME>* PROJECT**	**SAMPLE PROJECT**
Application Domain	Weather Satellite	Communications Satellite
Product Type	Ground System (Command and Control)	Ground System (Command and Control)
Acquisition Phase[2]	EMD	EMD
Software Development Phase(s)	Design; Coding and Unit Test	Design; Coding and Unit Test, Increments 1 and 2
Award Date		1/94
Contract Duration	8 Years	5 Years
Current Project Phase/ Contract Month[3]		EMD: Between System PDR and System CDR/Month 24
Prime/Subcontractors[4]	2 Software Subs	Prime & 1 Software Sub
Software KSLOC[5]	750	500
Language(s) and Percentages	Ada 95: 90% C++: 10%	Ada 83 77: 75% C++: 25 %
Target Processor(s)/OS(s)	RISC 6000/UNIX	VAX 6200/VMS 6.2
Applicable Standards	IEEE 1498	DoD-STD-2167A & 2168

[1]The customer information shall include the name, current address and phone number for one or more customers that the Government may contact with respect to contract performance.

[2]For "*<PROGRAM NAME>* PROJECT," phase(s) in which Critical Capability(ies) are to be used; for "SAMPLE PROJECT," phase in which sample was generated.

[3]The phase/month of the SAMPLE PROJECT as of the current date.

[4]Contractors developing the software products specified in the "Product Type" row

[5]Total number of KSLOC for software specified in the "Product Type" row

Figure 1. Project Sample Data Cover Sheet

Cover Sheet for Project's Sample Data		
Contractor:		
Sample Project Name:		
Sample Project Customer :		
Critical Capability(ies):		
Title of Sample:		
Explain why your experience on the sample project is relevant to the *<Program Name>* project.		
ATTRIBUTES	***<PROGRAM NAME>* PROJECT**	**SAMPLE PROJECT**
Application Domain		
Product Type		
Acquisition Phase[1]		
Software Development Phase(s)		
Award Date		
Contract Duration		
Current Project Phase/ Contract Month[2]		
Prime/Subcontractors[3]		
Software KSLOC[4]		
Language(s) and Percentages		
Target Processor(s)/OS(s)		
Applicable Standards		

[1]The customer information shall include the name, current address and phone number for one or more customers that the Government may contact with respect to contract performance.

[2]For "*<PROGRAM NAME>* PROJECT," phase(s) in which Critical Capability(ies) are to be used; for "SAMPLE PROJECT," phase in which sample was generated.

[3]The phase/month of the SAMPLE PROJECT as of the current date.

[4]Contractors developing the software products specified in the "Product Type" row

[5]Total number of KSLOC for software specified in the "Product Type" row

Figure 2. Blank Project Sample Data Cover Sheet

Capability Definition Matrix	Contractor Team A
Functional Area 2. Systems Engineering	**Critical Capability Area** 2.2 Computer System Architecture Design and Review Process

Critical Capabilities

2.2.1 Architecture Definition Q1	
Document Category Number. Category(ies)	**Location of Substantiating Information** Unique Id; Title; Document Location in SDCE response; Location in Referenced Document; Note
1. Current Contract	V6-3; SDP; Tab 3 in Volume 6; p 50, paragraphs 2.2.3 through 3.4; This document has been updated from the SDP used for Pre-EMD
2. Current Contract	V2-2; Draft IMP; Tab 2 in Volume 2; p 24-35, paragraph 4.1
3. Current Contract	V2-1; Draft IMS; Tab 1 in Volume 2; p 20-15, paragraph 3
4. Other Project	V6-3; Project X SEMP; Tab 3 of Volume 6; p 50-70, paragraphs 5 through 12
5. Other Project	V7-4; Project Y System Specification; Tab 4 of Volume 7; p 30-35, paragraphs 3.3.1 through 3.6 and Appendix 20.1, p 3-15, paragraphs 2 through 6, except paragraph 3
2.2.1 Architecture Definition Q2	
Document Category Number. Category(ies)	**Location of Substantiating Information** Unique Id; Title; Document Location in SDCE response; Location in Referenced Document; Note
1. Current Contract	V6-3; SDP; Tab 3 in Volume 6; p 50, paragraphs 2.2.3 through 3.4; Updated from Pre-EMD
2. Current Contract	V2-2; Draft IMP; Tab 2 in Volume 2; p 24-35, paragraph 4.1
3. Other Project	V8-3; Project Z SEMP; Tab 3 of Volume 8; p 50-70, paragraphs 5 through 12
4. Other Project	V7-4; Project Y System Specification; Tab 4 of Volume 7; p 30-35, paragraph 3.3.1
5. New; Company	V9-4; Company ABC Procedure 6, Review Procedure, Tab 4 of Volume 9; p 30-35, paragraph 3.3.1
2.2.2 Adequacy of Architecture Design Q1	
Document Category Number. Category(ies)	**Location of Substantiating Information** Unique Id; Title; Document Location in SDCE response; Location in Referenced Document; Note
1. Current Contract	V6-3; SDP; Tab 3 in Volume 6; p 60, paragraphs 4.2.3 through 5.4; Updated from pre-EMD
2. Current Contract	V2-1; Draft IMS; Tab 1 in Volume 2; p 20-15, paragraph 3
3. Corporate/Organization	V9-5; Company ABC Policy 5, Architecture Design Review; Tab 5 of Volume 9; p 20-30, paragraphs 2 to 8

Figure 3. Sample Capability Definition Matrix

INSTRUCTIONS

Document Category:

- Each document used as a reference for a given Critical Capability question should be individually numbered. There are four (4) categories of information: Current Contract (<*Program Name*> PD/RR), Other Project, Corporate/Organization, and New. These categories indicate the primary sources of the material in the referenced document. More than one category may be listed. New implies the substantiating information is referenced in the appropriate CC response.

Location of Substantiating Information

- The document reference must present its information in the following order: Unique Id, Title, Document Location, Location in response document, and Note. Of the five items only the first four (4) are required.
- Each item of substantiating information shall have its own unique identification that is used consistently in any reference.
- The Name/Title of each document must be unambiguous with respect to identifying the actual document referenced. In some cases exact titles, document and version numbers, and issue dates will all be necessary to uniquely identify the document intended. In other cases a set of initials may be sufficient.
- The Document Location must be unambiguous. In some cases such as a reference to a short extracted item, a volume number and tab within the volume is sufficient. In other cases the complete volume, tab, page, and paragraph references are required.
- The Location in Referenced Document of the information to be evaluated in the referenced document must be unambiguous. The actual material to be evaluated must be completely and unambiguously delineated by whatever methods are most meaningful for the referenced document. For example, in most documents both page numbers and paragraph numbers will be required to completely identify the appropriate material. If lower level delineation is required, for example at the non-numbered paragraph level, or if the form of the document does not lend itself to page and paragraph numbering to unambiguously identify the intended material, then the contractor is expected to develop and document an appropriate scheme and include it as part of the SDCE response.
- The Note in the document reference is not required. However, if it is used it should only contain information to help identify the appropriate content or to place the content in context for evaluation.

Figure 4. Capability Definition Matrix Instructions

Capability Definition Matrix	Contractor Team A
Functional Area	**Critical Capability Area**

Critical Capabilities

Document Category Number. Category(ies)	**Location of Substantiating Information** Unique Id; Title; Document Location in SDCE response; Location in Referenced Document; Note

Document Category Number. Category(ies)	**Location of Substantiating Information** Unique Id; Title; Document Location in SDCE response; Location in Referenced Document; Note

Document Category Number. Category(ies)	**Location of Substantiating Information** Unique Id; Title; Document Location in SDCE response; Location in Referenced Document; Note

Page ____of ____

Figure 5. Blank Capability Definition Matrix

Sample Data Inventory | **Contractor A**

Number	Unique Id	Title	Document Location	Critical Capability(ies) and Question(s)
•••				
12	V2-1	Draft IMS	Volume 2, Tab 1	2.2.1 Q1, Q2, 2.2.2 Q1, Q4
13	V2-2	Draft IMP	Volume 2, Tab 2	2.2.1 Q2
•••				
20	V4-4	Project X Software Development Metrics Report	Volume 4, Tab 4	4.4.2 Q1, Q2, Q3, Q4
•••				
25	V6-3	SDP	Volume 6, Tab 3	2.2.1 Q1, Q3, 2.2.2 Q1, Q3
•••				
33	V7-2	Company Process Policy 5	Volume 7, Tab 2	2.2.2 Q1, Q3
34	V7-3	Project X SEMP	Volume 7, Tab 3	2.2.1 Q4, Q5
35	V7-4	Project X System Specification	Volume 7, Tab 4	2.2.1 Q3
•••				

Page ____of ____

Figure 6. Example - Sample Data Inventory

Instructions

The Unique Id, Title and Document Location fields shall be the same as those used in the Capability Definition Matrix.

- Number — Each item in the Inventory shall be numbered individually.
- Unique Id: — Each item in the inventory shall have a unique identifier that is used consistently in references throughout the SDCE response.
- Title: — The title for each item shall be used consistently throughout the response.
- Document Location — This shall be the location in the delivered material for the item itself.

The Location in Referenced Document references shall be the same as those used in the Cover Sheet for Sample Data

- Critical Capabilities and Questions — Each item shall list the Critical Capabilities and questions that it supports.

Figure 7. Sample Data Inventory Instructions

Sample Data Inventory

Number	Unique Id	Title	Document Location	Critical Capability(ies) and Question(s)

Page ____of ____

Figure 8 Blank Sample Data Inventory

Appendix C—CMM Level-3-Equivalent Core Set of Questions and Criteria

In the following pages, there is a matrix showing the SDCE Critical Capability, the SDCE Criterion or Question, and the Key Process Area and Key Practice (KPA.KP) of the SW-CMM V1.1 to which the criterion or question maps. For example, Critical Capability 1.1.1 (Organizational Approach) is under Critical Capability Area 1.1 (Management Authority, Responsibility, and Accountability) under Functional Area 1 (Program Management); "SPP.AB2" represents Software Project Planning Ability 2 in the SW-CMM V1.1. Each Key Process Area has an abbreviation to those familiar with the SW-CMM, and these abbreviations are defined in the Acronym List.

The other SW-CMM Key Process Area abbreviations are as follows:

AB	Ability
AC	Activity
CO	Commitment
M	Measurement
V	Verification

Critical Capability	Criterion or Question	KPA.KP
1.1.1	**C1.** The software development and management functions are organized consistent with the proposed overall system development organizational structure (e.g., straight functional, Integrated Product Teams (IPTs)) and include identified support functions to the system engineering, subcontractor development and other functional development support activities as needed. **Q1 Q3**	SPP.AB2 SPP.AC1
1.1.1	**C3.** The software engineering organization is structured such that all program software (including support software) development is assigned to specific organizational elements. **Q4**	SPTO.AB2
1.1.1	**Q1.** Describe the total software development organization, top to bottom, including intermediate organizational supervisory levels. How is this software development function organizationally integrated and consistent with the program's overall system development organizational structure (e.g., straight functional, IPTs, etc.)? Describe the major software subcontractors' organizations to develop software. Describe any formal agreements between team members that define specific responsibilities for development. **C1**	SPP.AB2 SPP.AC1
1.1.1	**Q4.** Describe, within the identified software development organization and structure, the responsibility assignments for all program software including support, integration, and test software. **C3**	SPTO.AB2
1.1.2	**C3.** Project commitments and changes to commitments made to individuals and groups external to the organization are reviewed with senior management according to a documented procedure. Approved changes to commitments are communicated to affected groups. **Q7**	SPP.AC4 SPTO.AC3 SPTO.AC4
1.1.2	**Q7.** Describe how senior management reviews and approves commitments and changes to commitments made to groups and individuals external to the organization. Is there a documented procedure that describes these senior management reviews? How are approved changes to commitments communicated to affected groups? **C3**	SPP.AC4 SPTO.AC3 SPTO.AC4
1.2.1	**C1.** The program planning accounts for the integration of software development and management with system and hardware management. **Q1**	SPP.AC2 SPP.AC3

Critical Capability	Criterion or Question	KPA.KP
1.2.1	**C3.** The program planning includes the necessary reviews, accountability, status assessment, schedule control and reporting to manage the software related system development activities leading to the definition of the software requirements baseline. **Q2 Q3**	ISM.AC4 ISM.AC11 SPP.AC3 SPP.AC7 SPP.V2 SPTO.AC1
1.2.1	**C4.** The program planning includes a series of technical and management reviews with associated completion criteria that is used to control the development progress. **Q3**	IC.AC7 ISM.AC11 SPTO.AC12 SPTO.AC13 SPTO.AC9
1.2.1	**Q1.** How is your software development planning integrated with systems management and hardware management? **C1**	SPP.AC2 SPP.AC3
1.2.1	**Q2.** Describe your planning process used to establish the front-end software related system development activities. Describe your process to status and report these activities including specific criteria and control measures. Who is responsible to perform these front-end management activities? **C3**	SPP.V2
1.2.1	**Q3.** Describe your technical and management reviews used to control the development progress throughout the entire development period. Define these events and corresponding criteria. How are these events incorporated into the Systems Engineering Master Plan, Systems Engineering Master Schedule, Systems Engineering Detailed Schedule, and the Software Development Plan? **C3 C4**	IC.AC7 ISM.AC4 ISM.AC11 SPP.AC3 SPP.AC7 SPTO.AC1 SPTO.AC9 SPTO.AC12 SPTO.AC13
1.2.2	**C3.** The program has a mutually consistent and integrated statement of work (SOW), CWBS, work definition, scheduling, and cost tracking system and is used as the basis for program status and control. **Q6**	SPP.AB1
1.2.2	**Q6.** Describe how your CWBS procedures integrate with your statement of work (SOW), work definition process, scheduling process, and cost tracking system. Describe how the CWBS is used to support program status and control. **C3**	SPP.AB1
1.2.4	**C2.** The program's software scheduling and status system and proposed schedules are consistent and integrated with the Software Development Plan (SDP) and the program system level schedules, including the Systems Engineering Master Plan, Systems Engineering Master Schedule, and Systems Engineering Detailed Schedule (SEMP/SEMS/SEDS) as appropriate. **Q3 Q4 Q6**	SPP.AC2 SPP.AC3 SPP.AC7 SPTO.AC1

Critical Capability	Criterion or Question	KPA.KP
1.2.4	**Q3.** Describe how your process to establish software schedules integrates with the program's higher level scheduling system (e.g., SEMS and SEDS). **C2**	SPP.AC2 SPP.AC3 SPP.AC7
1.2.4	**Q6.** Describe how software schedules are defined, referenced, used, and updated in the SDP. **C2**	SPTO.AC1
1.3.1	**C1.** As part of the subcontractor selection process, documented procedures exist to evaluate subcontractors' capability and capacity to develop software. **Q1 Q2**	SSM.AC2
1.3.1	**Q1.** How are potential subcontractors' software development capabilities and capacities evaluated prior to selecting a specific subcontractor? **C1**	SSM.AC2
1.3.1	**Q2.** Where is this procedure for evaluating subcontractors' software capabilities and capacities documented? **C1**	SSM.AC2
1.3.2	**C3.** Periodic management and technical reviews to address subcontractor development progress are conducted and are reflected in the program's SDP/SEMP/SEMS/SEDS. **Q1**	SSM.AC9
1.3.2	**C4.** A process is defined to specify and control the subcontractor's performance requirements, interfaces, deliverables and product testing. **Q3**	SSM.AC1
1.3.2	**C5.** A documented process exists which requires reviewing and assessing the technical content of subcontractor generated design information and documentation. **Q4 Q5 Q6 Q7 Q8**	SSM.AC7 SSM.AC8 SSM.AC9 SSM.AC13
1.3.2	**C6.** The software test and verification process includes subcontractor developed software and incorporates the subcontractor software test and verification management and results into the overall hierarchical test process. **Q9**	SSM.AC12
1.3.2	**C9.** The prime/subcontractor contractual agreement is effectively used as the basis for managing the subcontract. **Q13**	SSM.AC3
1.3.2	**C10.** Changes to the subcontracted statement of work, subcontract terms and conditions and other commitments are resolved in accordance with a documented procedure. **Q14**	SSM.AC6
1.3.2	**Q1.** Fully describe your process for subcontractor management including reporting and control of the subcontractor software development activities. How does this process relate to and integrate with your overall system program management approach? Describe how the subcontractor management and review activities are reflected in the program level Systems Engineering Master Plan, Systems Engineering Master Schedule, and Systems Engineering Detailed Schedule (SEMP/SEMS/SEDS). **C1 C3**	SSM.AC9

Critical Capability	Criterion or Question	KPA.KP
1.3.2	**Q3.** How do you specify and control the subcontracted software technical/performance requirements, interfaces, deliverables, and product testing (test requirements and criteria)? **C4**	SSM.AC1
1.3.2	**Q4.** Describe your process for establishing and conducting periodic management and technical reviews and interchanges with your subcontractors. **C5**	SSM.AC7 SSM.AC8 SSM.AC9 SSM.AC13
1.3.2	**Q9.** What technical completion criteria for software are identified in the subcontract? Describe your test criteria and procedures for accepting subcontracted software. How is subcontracted software incorporated into your software integration and test process? **C6**	SSM.AC12
1.3.2	**Q13.** Describe how the prime/subcontractor contractual agreement is used as the basis for managing the subcontract. **C9**	SSM.AC3
1.3.2	**Q14.** Describe how changes to the subcontracted statement of work, subcontract terms and conditions, and other commitments between the prime contractor and the subcontractors are resolved. **C10**	SSM.AC6
1.3.3	**C3.** Subcontractor SDPs are reviewed and approved by the prime contractor. **Q3**	SSM.AC4 SSM.AC5
1.3.3	**C4.** Procedures ensure that the program's development standards and procedures are applied to subcontractor development efforts or a process is in place to ensure that subcontractor standards and procedures are used which are compatible with the program's development processes. **Q3 Q4**	SSM.AC10
1.3.3	**C6.** If award fees or incentives are established for subcontractor-developed software, measurable award fee or incentive criteria are established. **Q5**	SSM.AC13
1.3.3	**Q3.** Are the subcontractor SDPs reviewed and approved? How are these plans incorporated into your subcontractor development monitoring and tracking activity? **C3 C4**	SSM.AC4 SSM.AC5 SSM.AC10
1.3.3	**Q5.** Describe your approach to establishing award fees and incentives for subcontractor-developed software. Are predefined criteria established? Describe the nature of these criteria. Do you plan the use of award fees or incentives on this contract? **C6**	SSM.AC13
1.3.4	**C3.** The prime contractor's configuration management group follows an acceptable, documented procedure for monitoring software configuration management activities for all software development groups, including associate contractors and software subcontractors. **Q3**	SSM.AC11

Critical Capability	Criterion or Question	KPA.KP
1.3.4	**Q3.** Where are the procedures for monitoring software configuration management activities documented? Who monitors these activities? Does this procedure cover all software development groups, including the prime contractor, associate contractors and software subcontractors? **C3**	SSM.AC11
1.5.1	**C2.** Critical paths and tasks in the software development and associated schedules are identified and monitored. **Q4 Q5**	ISM.AC9
1.5.1	**Q4.** Identify the critical tasks and paths associated with the proposed development plan. Describe your process to monitor these critical elements. **C2**	ISM.AC9
1.5.2	**C1.** Risk management strategies required to identify and reduce risk are identified consistent with the program's cost, schedule, resource, and performance baselines. **Q1**	ISM.AC10 SPP.AC13 SPTO.AC10
1.5.2	**C2.** Identified risks are tracked and managed throughout the program development. **Q2**	ISM.AC10 SPP.AC13 SPTO.AC10
1.5.2	**Q1.** Describe your risk management process. What role will prototypes and demonstrations play in risk management? **C1**	ISM.AC10 SPP.AC13 SPTO.AC10
1.5.2	**Q2.** Describe how identified risk areas are analyzed, tracked, and monitored throughout the program development. **C2**	ISM.AC10 SPP.AC13 SPTO.AC10
2.1.1	**C5.** A defined process is used to generate the initial versions of the Software Requirements Specifications (SRS) and the Interface Requirements Specifications (IRS). A process to develop and review verification requirements for each performance requirement is in place. **Q5**	RM.AB2 SPE.AC2
2.1.1	**Q5.** Describe the process used to generate the SRS and IRS. Describe the process to define verification requirements for each performance requirement as part of the requirements and definition (specification preparation) process. **C5**	RM.AB2 SPE.AC2
2.1.2	**C2.** Software Requirements Specifications (SRS) and Interface Requirements Specifications (IRS) are analyzed and refined to assure that all requirements allocated to software are adequately addressed, and that they do not include inappropriate levels of design information. They are reviewed by all affected parties. **Q1 Q3**	RM.AC1
2.1.2	**Q3.** Describe the process by which the SRS and IRS are analyzed and refined to assure that all requirements allocated to software are adequately addressed. **C2**	RM.AC1

Critical Capability	Criterion or Question	KPA.KP
2.1.3	**C2.** All changes to requirements, including those generated by the customer, are managed by means of a defined change process. **Q2**	RM.AC3
2.1.3	**C3.** The Allocation of new and additional requirements between hardware and software is managed by a structured change process; the reallocation of existing requirements between hardware and software is managed by a structured change process. **Q3**	RM.AC3
2.1.3	**Q2.** Describe the requirements change control process, with reference to both internally and externally generated changes. **C2**	RM.AC3
2.1.3	**Q3.** What process is used to control the allocation of changed (new or existing) requirements between hardware and software? **C3**	RM.AC3
2.1.4	**C1.** The structured change process for requirements assures that the software impact for each proposed change is addressed. **Q1**	RM.AC3
2.1.4	**Q1.** How is the software impact for proposed changes to system requirements addressed? **C1**	RM.AC3
2.4.1	**C1.** Throughout the development life-cycle there is periodic coordination among developers, acquisition organizations, users, maintainers, and testers regarding user needs, acquisition organization resources, technology status, and system requirements. Requirements changes that result from interaction with users, maintainers, and testers are managed with acquisition organization approval. **Q1 Q2 Q3 Q4**	IC.AC1
2.4.1	**C2.** There is a systems engineering process which (as appropriate) emphasizes an integrated product development approach, and which defines the systems engineering interfaces with the other engineering disciplines and development activities, as well as interfaces between the system and subsystem developers. **Q5 Q6 Q7**	IC.AC2
2.4.1	**C3.** A process exists to manage, provide an escalation path for, and resolve conflicts regarding intergroup issues, including system-level issues that arise internally or with subcontractors. **Q8 Q9**	IC.AC6
2.4.1	**C4.** Critical dependencies between development groups are identified and tracked. **Q10 Q11 Q12**	IC.AC4
2.4.1	**Q1.** Describe the processes to be followed to have users' and maintainers' needs and viewpoints adequately reflected in system requirements throughout the development. **C1**	IC.AC1

Critical Capability	Criterion or Question	KPA.KP
2.4.1	**Q2.** Describe the processes to be followed to keep system requirements in balance with acquisition organization resources throughout the development. **C1**	IC.AC1
2.4.1	**Q5.** To what extent is an integrated product development approach to be followed? **C2**	IC.AC2
2.4.1	**Q7.** How will interfaces between the various system and subsystem developers be managed? **C2**	IC.AC2
2.4.1	**Q8.** Describe the processes for conflict resolution to be used internally between development groups. **C3**	IC.AC6
2.4.1	**Q9.** Describe the processes for conflict resolution to be used between prime contractors and subcontractors, and between subcontractors. Describe the processes used to identify and resolve intergroup product interface issues. **C3**	IC.AC6
2.4.1	**Q11.** Describe the processes for identifying new critical dependencies during development. **C4**	IC.AC4
2.4.1	**Q12.** How are critical dependencies between development groups tracked? **C4**	IC.AC4
2.4.2	**C1.** The support tools used by the different engineering groups enable effective communication and coordination. **Q1**	IC.AB2
2.4.2	**Q1.** Where different development groups have an interface, what support tools will be used to communicate and share data? Describe any areas of potential difficulty. **C1**	IC.AB2
2.5.2	**C1.** Software engineering coordinates with systems engineering on all items that flow down to software engineering; for example, the system architecture, information models, and identification, definition, and allocation of software requirements. Approved changes to the program baseline that effect the software development are communicated with software engineering and support groups such as SQA and SCM. **Q1**	IC.AC5 RM.AC1 SPP.AC1
2.5.2	**Q1.** Describe the role of software engineering on items that flow down from systems engineering to software engineering, such as system architecture, information models, and the identification, definition, and allocation of software requirements. How are approved changes to the program baseline that effect the software development communicated with software engineering and support groups? **C1**	IC.AC5 RM.AC1 SPP.AC1
2.5.3	**C1.** Systems engineering milestones (including formal reviews) are defined and implemented with clear completion criteria in the SEMP/SEMS. **Q1**	SPTO.AC13

Critical Capability	Criterion or Question	KPA.KP
2.5.3	**Q1.** Describe the intended use of SEMP/SEMS/SEDS on the program. Are all major software milestones addressed in the SEMP, SEDS and SEMS? Are completion criteria specified with all events? **C1**	SPTO.AC13
2.6.1	**C2.** Test planning for each system build includes the multiple levels of system integration and test (from units to Computer Software Configuration Items (CSCIs) to subsystem to system-level test). **Q2**	SPE.AC7
2.6.1	**Q2.** If system builds are planned, describe how test planning for each system build includes the multiple levels of system integration and test (from units to CSCIs to subsystem to system-level test). **C2**	SPE.AC7
3.1.1	**C1.** Estimates for size, effort, cost, and schedule of each software component are generated according to a documented procedure. Estimates for incrementally developed software are generated consistently with published methods and company experience. **Q1**	ISM.AC6 ISM.AC7 SPP.AC9 SPP.AC10 SPP.AC12
3.1.1	**C3.** Estimates of the required critical computer resources needed by each of the software components are generated according to a documented procedure. **Q3**	ISM.AC8 SPP.AC11
3.1.1	**C7.** All data required to repeat the above estimates for each software component are recorded and maintained. **Q6** **Q7**	SPP.AC15 SPTO.AC11
3.1.1	**Q1.** How are estimates for the size, effort, cost, and schedule of each software component generated? Which published estimating methods and models are used? Describe how estimates are developed for any planned incremental development or release. Describe your experience with this method relative to actual size, effort, cost, and schedule of completed projects. **C1**	ISM.AC6 ISM.AC7 SPP.AC9 SPP.AC10 SPP.AC12
3.1.1	**Q3.** How are estimates generated for required critical computer resources needed by each software component? How are computer resources estimated and balanced across the program to ensure that critical needs are met? **C3**	ISM.AC8 SPP.AC11
3.1.1	**Q6.** How is the data required to repeat the above estimates for each software component recorded and maintained? Is the data configuration controlled and available to all who need it? Are occasional audits done to verify that the required data is accurate and available? **C7**	SPP.AC15 SPTO.AC11
3.1.1	**Q7.** Who has the responsibility for development and storage of the above estimates? Who ensures that estimates are done according to procedure, and that the data is recorded and maintained? **C7**	SPP.AC15

Critical Capability	Criterion or Question	KPA.KP
3.1.2	**C1.** Software components and work packages of manageable size and development effort are defined to enable management of the entire software system. **Q1**	SPP.AC5
3.1.2	**C2.** The defined set of software work packages is used to manage the work tasks associated with software development. **Q2**	ISM.AC4
3.1.2	**Q1.** How is the overall software effort organized into manageable software components? What factors are considered in determining the appropriate size and development effort for each of the components? How is the software organization documented? **C1**	SPP.AC5
3.1.2	**Q2.** How are software work packages planned and defined? Describe the criteria for acceptable software work packages. Explain how the software work package is used to manage the work (i.e., plan, define, assign resources and responsibility, status and report progress). **C2**	ISM.AC4
3.1.3	**C4.** A well-defined Software Development Process Model has been selected for use on the program at hand. **Q4**	SPP.AC5
3.1.3	**Q4.** Describe the Software Development Process Model selected for the program: what activities it comprises, how activities are sequenced and iterated, what are the entrance and exit criteria from one activity to the next, and from one iteration to the next. **C4**	SPP.AC5
3.1.4	**C4.** All of the involved parties within the software development organization participate in the generation of the SDP, and demonstrate understanding and commitment to its terms. The SDP is coordinated throughout the program organization, including subcontractors. **Q4**	SQA.AC3
3.1.4	**C9.** The SDPs are developed and maintained using a sound and complete process. **Q9**	IC.AC3 ISM.AC3 SPP.AC6 SPP.AC7 SPTO.AB1 SPTO.AC2
3.1.4	**C10.** A process exists for coordinating SDPs across team members and ensuring integrity and supportability of the program's software. **Q10 Q11**	IC.AC3
3.1.4	**C11.** The allocated requirements form the basis for the SDP, work products and activities. **Q12**	RM.AC2

Critical Capability	Criterion or Question	KPA.KP
3.1.4	**Q4.** How do all of the components of the software development organization participate in generating the SDP? How do they demonstrate understanding and commitment to the terms of the SDP? Which organizations, including subcontractors, coordinate on the SDP? How are the terms, dependencies, and responsibilities negotiated and communicated, both internal to the prime and among the various subcontractor? **C4**	SQA.AC3
3.1.4	**Q9.** Describe the process used to generate the SDPs for the project, who participates in the effort, where results are recorded, and how plans are maintained. Describe the software development planning that will be performed; what software development life-cycle activities will be covered in SDPs; how the plan accounts for processes, schedules, and manpower needed to develop the software to be delivered. **C9**	IC.AC3 ISM.AC3 SPP.AC6 SPP.AC7 SPTO.AB1 SPTO.AC2
3.1.4	**Q11.** Which components of the software development organizations (across the prime contractor and subcontractors) coordinate on the final contents of the SDPs and how do they do that? How are terms, dependencies, and responsibilities negotiated and communicated internal to the prime contractor, between the prime contractor and subcontractors, and among subcontractors? **C10**	IC.AC3
3.1.4	**Q12.** Describe how the allocated requirements drive the development of the SDP, work products to be created and the development activities established to complete the work products? **C11**	RM.AC2
3.2.1	**C1.** The size, effort, cost, and schedule status of each of the software work packages is periodically measured and reviewed by engineering management and corrective actions are taken when pre-established variance thresholds are exceeded. **Q1**	ISM.AC5 ISM.AC7 SPTO.AC12 SPTO.AC5 SPTO.AC6 SPTO.AC8 SPTO.AC9
3.2.1	**C2.** The critical computer resources required by and allocated to each of the software work packages are periodically measured and reviewed by management, and corrective actions are taken when pre-established variance thresholds are exceeded. **Q2**	ISM.AC8 SPTO.AC7

Critical Capability	Criterion or Question	KPA.KP
3.2.1	**Q1.** How often will engineering and program management measure and review the size, effort, cost, and schedule status of each of the software components? What criteria and conditions will trigger corrective actions? How will the success of the corrective actions be measured? What provisions exist for event-driven engineering management reviews? **C1**	ISM.AC5 ISM.AC7 SPTO.AC12 SPTO.AC5 SPTO.AC6 SPTO.AC8 SPTO.AC9
3.2.1	**Q2.** How often will critical computer resources required by each of the software components be measured and reviewed by engineering and program management? When will it be deemed necessary to take corrective action? Who is responsible for setting variance thresholds? **C2**	ISM.AC8 SPTO.AC7
3.2.2	**C1.** The status of each software work package is reported to all involved levels of engineering and program management through periodic reporting up the chain of command. **Q1**	ISM.AC11 SPTO.AC9
3.2.2	**C2.** Development process/performance and product quality measurements are recorded, analyzed, and used for improving process and product quality on the program. These data are recorded and maintained for organizational process and product quality improvements. **Q2**	OPD.AC5 SPTO.AC11
3.2.2	**Q1.** How is the status of each software work package reported up the chain of command? What specific elements of software status (e.g., units, components, configuration items, subsystem, system) are reported to each management level from first-level supervisor through program manager? What situation, condition, or threshold would trigger a status report to a higher level of management than normally would be necessary for a work package? **C1**	ISM.AC11 SPTO.AC9
3.2.2	**Q2.** What actual measurements of development performance and product quality will be recorded during software development? How will these measurements be analyzed and used for changing and improving products and processes? How will metrics be recorded and maintained? Who is responsible for the collection, storage, and analysis of metrics? **C2**	OPD.AC5 SPTO.AC11
3.3.1	**C1.** The software requirements are analyzed for completeness, correctness, clarity, feasibility and verifiability. **Q2 Q3**	RM.AC1 SPE.AC2
3.3.1	**C2.** Requirements that are derived from the Software Requirements Specification are documented and maintained. **Q4**	RM.AB2 SPE.AC2

Critical Capability	Criterion or Question	KPA.KP
3.3.1	**Q2.** What are the software requirements analyzed for, i.e., completeness, correctness, etc.? How do you determine that the software requirements are complete, adequate, and verifiable? **C1**	RM.AC1 SPE.AC2
3.3.1	**Q4.** If additional requirements are derived from the baselined requirements, where are they documented? How are they maintained? How is their impact on cost and schedule determined? **C2**	RM.AB2 SPE.AC2
3.3.2	**C1.** Software development artifacts (requirements, design, code and documentation) are revised as changes to the requirements are incorporated. **Q1**	RM.AC2 SPE.AC2
3.3.2	**C2.** As changes and additions to the requirements are incorporated, the Software Development Plans (SDPs) and program baselines (cost and schedule) are reviewed and modified if necessary. **Q2**	RM.AC2 RM.AC3
3.3.2	**Q1.** Describe the software development activities resulting from a change in or addition to the requirements. When do they get performed? How do you ensure that they are performed? **C1**	RM.AC2 SPE.AC2
3.3.2	**Q2.** Describe the software planning activities that result from a change in the requirements. When are they performed? How do you ensure that they are performed? **C2**	RM.AC2 RM.AC3
3.4.1	**C1.** A methodology is used to develop, document and maintain the top-level and detailed software design. **Q1**	SPE.AC3
3.4.1	**C6.** The selected design methodology is compatible with other methodologies adopted on the program. **Q9 Q10**	SPE.AC3
3.4.1	**Q1.** Describe the process and specific methodologies used to develop the top-level and detailed software design. Is the same methodology used to maintain the design through development and life cycle support? What tools are used to support the methodology? **C1**	SPE.AC3
3.4.1	**Q9.** Is the design methodology compatible with the requirements analysis methodology? Is it compatible with the development language? **C6**	SPE.AC3
3.4.2	**C3.** Two-way traceability between design and the requirements is established and maintained. **Q5**	SPE.AC3
3.4.2	**Q5.** How is traceability established from the requirements to the design, and from the design to the requirements? At what point in the design is it done, and by whom? How is it documented? How is it maintained? What tools are used? Is this traceability part of the exit criteria described above? **C3**	SPE.AC3

Critical Capability	Criterion or Question	KPA.KP
3.5.1	**C3.** The developed software is unit tested. Realistic resources and schedules are allocated to this level of testing. Units are tested in all increments of development. **Q7 Q8**	SPE.AC4
3.5.1	**C4.** The software is reviewed against the design, and 2-way traceability between the software code and the design is established and maintained. **Q9**	SPE.AC4
3.5.1	**C5.** Exit criteria exist for establishing that each lowest-level software unit has been implemented correctly, is performance tested and is in conformance with the coding standards. **Q10**	SPE.AC4
3.5.1	**Q7.** What processes and procedures are used to ensure that the design is implemented completely and correctly? At what component level? Who has that responsibility? **C3**	SPE.AC4
3.5.1	**Q9.** How is traceability from the software code to the design and from the design to the code established and maintained? When is it done? How is it documented and maintained? What tools are used? Who has that responsibility? **C4**	SPE.AC4
3.5.1	**Q10.** What exit criteria exist for establishing that each lowest-level software unit is ready for integration? Do they include compliance with coding standards? Do they include peer reviews? Do they include unit testing? Do they include conformance to the design? How are they enforced? **C5**	SPE.AC4
3.6.1	**C1.** The software integration planning takes into account the interdependencies between the different software components and the criticality of each component. **Q1 Q2 Q3**	SPE.AC6
3.6.1	**C2.** The software integration planning takes into account the availability of other components of the system. **Q1 Q4**	SPE.AC6
3.6.1	**C4.** The software integration planning and process accommodate software integration starting with the lowest level elements, i.e., units through all levels, including CSCI and CSCI/HWCI. **Q1**	SPE.AC6
3.6.1	**Q1.** Describe your process for planning the software integration. How many different components do you integrate at once? How do you determine the order for integrating the different software components? Describe how your integration process accommodates all levels of software integration. **C1 C2 C4**	SPE.AC6
3.6.1	**Q2.** How are the dependencies between the different software components determined? At what level? How does it affect integration planning? **C1**	SPE.AC6

Critical Capability	Criterion or Question	KPA.KP
3.6.1	**Q3.** How is the criticality of each component determined? What role does it play in integration planning? **C1**	SPE.AC6
3.6.2	**C1.** The software test process includes development of test plans, procedures, and test cases. **Q1**	SPE.AC5
3.6.2	**C3.** An approach is used that plans for all levels of testing to ensure thorough testing of the software. **Q3 Q4**	SPE.AC5 SPE.AC7
3.6.2	**C5.** A regression test methodology ensures that system performance is maintained after revisions are made to the software components. **Q7 Q8**	SPE.AC5
3.6.2	**Q1.** How are test plans, test procedures and test cases developed? When? By whom? Where are they documented? How are they reviewed? How are they controlled? **C1**	SPE.AC5
3.6.2	**Q3.** Does your software test and verification process define specific levels of software test? What are they? How do they relate to the structure of your software design? **C3**	SPE.AC5 SPE.AC7
3.6.2	**Q4.** What are the completion criteria for each level of testing? Do you generate test plans, and test procedures for each level? If so, how are they coordinated across the different levels? **C3**	SPE.AC7
3.6.2	**Q7.** What is your process for regression testing? Are there guidelines for when and how the regression tests should be run? Is regression testing factored into the schedules? **C5**	SPE.AC5
4.1.1	**C5.** The plan identifies methods for analyzing the program's quality measurements, for evaluating whether they meet the customer's needs, and for determining the necessary corrective actions. **Q7**	SQA.AC8
4.1.1	**Q7.** Does the quality plan describe how the quality data is analyzed, and how it is used? **C5**	SQA.AC8
4.2.1	**C1.** An organization is assigned the responsibility to monitor the software development process and the software products. **Q1**	SQA.AC3
4.2.1	**C2.** The responsibilities, mission, and interface(s) of SQA with the engineering, configuration management, and test functions are defined and documented. **Q1 Q2 Q3**	SQA.AC3 SQA.AC6 SQA.AC7 SQA.AC8
4.2.1	**C3.** The SQA group is empowered to effect changes to the program when quality goals are not followed. **Q4 Q5**	SQA.AC7
4.2.1	**Q1.** Describe the responsibilities of the SQA organization and how SQA interfaces with other organizations. **C1 C2**	SQA.AC3 SQA.AC8
4.2.1	**Q2.** Does the SQA organization communicate the results of SQA activities to the engineering organization? **C2**	SQA.AC6
4.2.1	**Q3.** How does the SQA function interface with engineering, configuration management, and test functions? **C2**	SQA.AC3 SQA.AC7

Critical Capability	Criterion or Question	KPA.KP
4.2.1	**Q4.** What can SQA organization do if the software development process and procedures are not being followed? **C3**	SQA.AC7
4.2.1	**Q5.** What mechanisms and channels exist for SQA to surface quality problems and elevate them in the management chain until they are resolved? **C3**	SQA.AC7
4.2.3	**C1.** The program follows a written SQA plan for measuring and monitoring the performance of the program's defined software process. **Q1**	SQA.AC1 SQA.AC2 SQA.AC7
4.2.3	**C2.** Adherence to the defined software development and management processes is verified. **Q2**	SQA.AC4
4.2.3	**C3.** SQA audits designated software work products to verify compliance with quality goals and adherence to the applicable standards and requirements. **Q3 Q4**	SQA.AC4 SQA.AC5
4.2.3	**Q1.** Where are SQA activities defined for the program? **C1**	SQA.AC1 SQA.AC2 SQA.AC7
4.2.3	**Q2.** Describe how SQA ensures compliance of the software development activities with the defined processes. Which processes are audited? How often? **C2**	SQA.AC4
4.2.3	**Q3.** Describe how SQA ensures compliance of the software management activities with the planned processes. Which processes are audited? How often? **C3**	SQA.AC4
4.2.3	**Q4.** Describe how SQA verifies that the software products adhere to the program's requirements, standards, and quality goals. **C3**	SQA.AC5
4.3.2	**C3.** Data on defects identified in peer reviews, document review and testing are collected and analyzed. **Q5**	SPE.AC5 SPE.AC9
4.3.2	**Q5.** Identify your approach to collecting defects resulting from peer reviews, testing, and design reviews. Is this approach contained in the quality plan? **C3**	SPE.AC5 SPE.AC9
4.4.1	**C3.** The established metrics process includes the requirements to define variance thresholds, which when broken, require corrective action. **Q4**	ISM.AC6 ISM.AC7 ISM.AC8
4.4.1	**Q4.** Describe your use of variance thresholds. Describe how these thresholds are established and used in development management. **C3**	ISM.AC6 ISM.AC7 ISM.AC8
4.5.1	**C1.** Internal documents exist that: identify required participants in the reviews, provide specific criteria for successful completion, and describe documentation required for the review and describe how follow-on actions are documented, tracked, and controlled. **Q1**	PR.AC1 PR.AC2

Critical Capability	Criterion or Question	KPA.KP
4.5.1	**C2.** Peer reviews are planned consistent with the peer review internal standards and procedures. **Q2**	PR.AC1
4.5.1	**C3.** Peer review plans specify the schedule of peer reviews. **Q2**	PR.AC1
4.5.1	**Q1.** Describe the documented internal peer review procedures and requirements including definition of required participants, completion criteria and review content, and follow-on action item resolution. **C1**	PR.AC1 PR.AC2
4.5.1	**Q2.** Describe how peer reviews are planned and scheduled. Describe how the peer review schedule is consistent with other program schedules (e.g., SEMP/SEMS). **C2 C3 C4**	PR.AC1
4.5.2	**C1.** Peer reviews are performed according to the peer review plan. **Q1**	PR.AC2
4.5.2	**C2.** Reviews are documented (i.e., review process, requirements, conduct, and results). **Q2**	PR.AC2 PR.AC3
4.5.2	**Q1.** Describe how peer reviews are performed according to the peer review plan. **C1**	PR.AC2
4.5.2	**Q2.** Describe how peer review results are documented and to whom results are distributed. **C2 C3**	PR.AC2 PR.AC3
4.7.1	**C2.** A process exists for the development, maintenance, and distribution of the program's SCM plan, standards and procedures. **Q3**	SCM.AC1
4.7.1	**C3.** An approved SCM plan is used as the basis for performing the SCM activities. **Q4**	SCM.AC1 SCM.AC2
4.7.1	**C4.** The SCM Planning requires creation and management of the program's software baseline library. The baseline library contains the functional, allocated, developmental and product baselines. **Q6**	SCM.AC3
4.7.1	**Q3.** What guidance exists for the development, maintenance and distribution of the program's SCM plan, standards, and procedures? **C2**	SCM.AC1
4.7.1	**Q4.** Is there a software configuration plan for this program? Who reviews and approves the plan? **C3**	SCM.AC1 SCM.AC2
4.7.1	**Q6.** Does the CM plan require creation and management of a program software baseline library? Where are the library procedures documented? **C4**	SCM.AC3

Critical Capability	Criterion or Question	KPA.KP
4.7.2	**C3.** Products from the software baseline library are created and released according to a procedure. Procedures exist for management of • software requirements document(s) • software design document • code • test plans, test procedures, and test cases • test results **Q4**	SCM.AC7
4.7.2	**C10.** The software work products to be placed under SCM are identified, including baselines and test configurations for each block and/or build called for by the developer's software development process model. **Q11**	SCM.AC4 SPP.AC8
4.7.2	**Q4.** Explain the SCM library procedures for documentation release, software release and (if applicable) "promoting" to another library. Identify the internal documents where these library procedures are documented, formally or informally. **C3**	SCM.AC7
4.7.2	**Q11.** Which products of the software development process will be placed under configuration control? **C10**	SCM.AC4 SPP.AC8
4.7.3	**C1.** Procedures and criteria are provided for a complete configuration audit including assigned responsibility. **Q1 Q2 Q3**	SCM.AC10 SCM.V3
4.7.3	**C2.** Software baseline audits are conducted. **Q1**	SCM.AC10 SCM.V3
4.7.3	**Q1.** Who is responsible to perform and approve the configuration audits? **C1 C2**	SCM.AC10 SCM.V3
4.7.3	**Q3.** What procedure(s) are followed when performing software audits? **C1**	SCM.AC10 SCM.V3
4.7.4	**C2.** Changes to baselines are controlled. **Q2 Q5**	SCM.AC5 SCM.AC6
4.7.4	**C4.** Change requests and problem reports for all configuration items/units are initiated, recorded, reviewed, approved, and tracked. **Q4**	SCM.AC5
4.7.4	**C5.** Status accounting (status of configuration items/units) is recorded. **Q5**	SCM.AC8
4.7.4	**C7.** Change control procedures, which include the equivalent of configuration control boards for software, are defined and integrated into the program change management process. **Q7**	SCM.AB1
4.7.4	**C10.** Software configuration management activities and the contents of software baselines are documented and standard reports are made available to affected groups and individuals. **Q10**	SCM.AC9

Critical Capability	Criterion or Question	KPA.KP
4.7.4	**Q2.** Describe the procedures the program follows to control changes to configuration items. **C2**	SCM.AC5 SCM.AC6
4.7.4	**Q4.** Does the configuration management process include configuration status accounting? **C1 C4**	SCM.AC5
4.7.4	**Q5.** How is status accounting achieved? Is the function automated? Describe the tools and process. **C2 C3 C5**	SCM.AC8
4.7.4	**Q7.** Describe the change control procedures to be followed for changes requested to products of the software development process under configuration control. **C7**	SCM.AB1
4.7.4	**Q10.** Explain how software configuration management activities and the contents of software baselines are documented and reported. Which individuals or groups receive or have access to this information? **C10**	SCM.AC9
4.8.1	**C3.** The documentation used to operate and maintain the software is developed and maintained consistently with the current software baseline. **Q3**	SPE.AC8
4.8.1	**Q3.** How is documentation developed and maintained? What process(es) assures accuracy and completeness? **C3**	SPE.AC8
4.8.2	**C5.** Consistency and currency is maintained across software work products including the software plans, process descriptions, allocated requirements, software requirements, software design, code, test plans, and test procedures. **Q5**	SPE.AC10
4.8.2	**Q5.** Is consistency maintained across software products from requirements through acceptance testing (i.e., traceability across software requirements, software plans, design, code, and test)? What ensures that this is accomplished? **C5**	SPE.AC10
5.1.1	**C1.** The organization's system and software development standards comprehensively describe the system and software development, their interfaces, and interdependencies. The standards also document the interfaces within and among the various system software and other disciplines. **Q1 Q2**	OPD.AC2
5.1.1	**C2.** The organizational standards provide a set of system and software engineering development models (e.g., waterfall, event-driven) for selection and use by the program. The descriptions of these models are compatible with the organization's standard system and software development process(es). **Q3**	OPD.AC3
5.1.1	**C4.** The organization's system development and software development process(es) standards are placed under configuration control. **Q5**	OPD.AC1

Critical Capability	Criterion or Question	KPA.KP
5.1.1	**Q1.** In your organization's system development and software development process(es) standards, how are activities and events described (e.g., inputs, outputs, readiness and completion criteria)? How are the relationships (sequencing, interfacing, and interdependencies) of the activities described? **C1**	OPD.AC2
5.1.1	**Q3.** Identify the system development and software development models (e.g., waterfall, event-driven) and explain how these are defined in your standards. How is compatibility between the organization's standard system development and software development process maintained and ensured? **C2**	OPD.AC3
5.1.1	**Q5.** Describe your approach for version control and controlling changes to the organization's standard system development and software development process(es). How do you know which version of the organization's standard is in use at a given time? How are changes to the standard assessed, incorporated within the standard, and incorporated by the program? **C4**	OPD.AC1
5.1.2	**C1.** A waiver procedure and tailoring guidelines and criteria are available to facilitate tailoring the organization's standard systems development software development process(es) to meet specific program requirements and needs. **Q1 Q2**	ISM.AC1 OPD.AC4
5.1.2	**Q1.** Describe any documented guidelines provided for tailoring organizational standards to specific program requirements. What specific program needs require tailoring on this program? How was the specific system development and software development model for this program selected? Given the systems and software development model for this program, how are the organization's system and software development processes and procedures tailored to be compatible with and support the development model? **C1**	ISM.AC1 OPD.AC4
5.1.2	**Q2.** Describe the procedure for waiving compliance with the organization's standard system development and software development process(es). How does it support application of the tailoring guidelines? Describe how the procedure provides flexibility for those cases where particular program needs require extensive tailoring. **C1**	ISM.AC1 OPD.AC4

Critical Capability	Criterion or Question	KPA.KP
5.1.3	**C1.** Past use data for standard organizational and program processes is collected. These data include estimates and actuals, quality measurements, peer review/test coverage and efficiency, number and severity of defects found. These experience-based data are made available to programs for planning and managing new programs. **Q1**	ISM.AC5 OPD.AC5 OPF.AC4
5.1.3	**C2.** A library of process-related documentation (e.g., program standards, measurement plans, process training materials) is maintained and made available to the program to support reuse of proven processes and interpretation of usage data. **Q2**	OPD.AC6 OPF.AC4
5.1.3	**Q1.** Explain how data from use of the organization's and programs' development processes and resulting products is collected and made accessible to the program for use in planning and managing its effort. In addition to the actual measurement data, what kind of related information is maintained to help the program understand and interpret the measurement data and assess it for reasonableness and applicability? **C1**	ISM.AC5 OPD.AC5 OPF.AC4
5.1.3	**Q2.** For the program, what kinds of process-related documentation is maintained and made available to support reuse of proven processes and interpretation of usage data? How are these documentation items catalogued for easy access? **C2**	OPD.AC6 OPF.AC4
5.1.4	[For each of the Process Institutionalization Critical Capability criteria/questions below, responses should be provided in a context that ensures coverage of at least the following software process areas: • Requirements management • Software project planning, tracking, oversight, and integrated software management • Software subcontract management • Software quality assurance • Software configuration management • Organizational process development and maintenance • Organizational training program • Software product engineering (requirements analysis, design, code, test and verification) • Intergroup coordination • Peer reviews A single response may be provided to each question, to the extent that institutionalization is treated similarly for all these process areas. If institutionalization is accomplished differently for one or more of these areas, the differences should be distinguished in the response.]	

Critical Capability	Criterion or Question	KPA.KP
5.1.4	**C1.** Senior management demonstrates commitment to the organizational software processes through written policy statements that mandate their use across projects and the organization, as applicable. **Q1**	IC.CO1 ISM.CO1 OPD.CO1 OPF.CO1 PR.CO1 RM.CO1 SCM.CO1 SPE.CO1 SPP.CO2 SPTO.CO2 SQA.CO1 SSM.CO1 TP.CO1
5.1.4	**C2.** Responsibility and authority is assigned to those responsible for implementing or providing leadership of the software process areas. The organizational structure provides for assignment or delegation of the roles and responsibilities necessary for implementing the activities performed within these areas. **Q2**	RM.AB1 SCM.AB2 SPP.AB2 SPP.CO1 SPTO.CO1 SQA.AB1 SSM.CO2 TP.AB1
5.1.4	**C3.** Adequate resources and funding are provided for performing the software process areas, including access to special skills or tools. **Q3**	IC.AB1 ISM.AB1 OPD.AB1 OPF.AB2 PR.AB1 RM.AB3 SCM.AB3 SPE.AB1 SPP.AB3 SPTO.AB3 SQA.AB2 SSM.AB1 TP.AB2

Critical Capability	Criterion or Question	KPA.KP
5.1.4	**C4.** A curriculum of required training in software processes is defined and provided for software developers, software project managers, and affected groups (e.g., systems engineering, program management, configuration management, quality assurance, process engineering). The content of this curriculum, which may differ according to assigned areas of responsibility, includes training or orientation, as applicable, in relevant software process areas. A combination of formal and informal vehicles is used for transferring skills and knowledge to the individuals in the organization. **Q4**	IC.AB2 IC.AB3 IC.AB4 IC.AB5 ISM.AB2 ISM.AB3 OPD.AB2 OPF.AB3 OPF.AC6 PR.AB2 PR.AB3 RM.AB4 SCM.AB4 SCM.AB5 SPE.AB2 SPE.AB3 SPP.AB4 SPTO.AB4 SPTO.AB5 SQA.AB3 SQA.AB4 SSM.AB2 SSM.AB3 TP.AB3 TP.AB4
5.1.4	**C5.** Measurements are made and used to determine the status and effectiveness of the activities performed for each software process area, at the project and/or organizational level as applicable. **Q5**	IC.M1 ISM.M1 OPD.M1 OPF.M1 PR.M1 RM.M1 SCM.M1 SPE.M1 SPE.M2 SPP.M1 SPTO.M1 SQA.M1 SSM.M1 TP.M1 TP.M2

Critical Capability	Criterion or Question	KPA.KP
5.1.4	**C6.** Management insight into the software process area activities is provided through periodic reviews with senior management. Activities for project processes are reviewed on both a periodic and event-driven basis with the project manager. **Q6**	IC.V1 IC.V2 ISM.V1 ISM.V2 OPD.V1 OPF.V1 RM.V1 RM.V2 SCM.V1 SCM.V2 SPE.V1 SPE.V2 SPP.V1 SPP.V2 SPTO.V1 SPTO.V2 SQA.V1 SQA.V2 SSM.V1 SSM.V2 TP.V1 TP.V2
5.1.4	**C7.** Adherence to defined software processes is verified objectively. An independent function, such as software quality assurance, conducts reviews and/or audits of software process area activities and work products and reports the results. **Q7**	IC.V3 ISM.V3 OPD.V1 PR.V1 RM.V3 SCM.V4 SPE.V3 SPP.V3 SPTO.V3 SSM.V3 TP.V3
5.1.4	**Q1.** Describe your written organizational policies that demonstrate senior management commitment to the standard software processes and mandate their use. **C1**	IC.CO1 ISM.CO1 OPD.CO1 OPF.CO1 PR.CO1 RM.CO1 SCM.CO1 SPE.CO1 SPP.CO2 SPTO.CO2 SQA.CO1 SSM.CO1 TP.CO1

Critical Capability	Criterion or Question	KPA.KP
5.1.4	**Q2.** Who is responsible for ensuring the software process areas are implemented? How are the activities assigned or delegated to those responsible for doing the work? Where are these roles and responsibilities documented? **C2**	RM.AB1 SCM.AB2 SPP.AB2 SPP.CO1 SPTO.CO1 SQA.AB1 SSM.CO2 TP.AB1
5.1.4	**Q3.** How is the adequacy of resources and funding for implementation of the software process areas determined? How are special skills or tools that are necessary for effective implementation identified and provided? **C3**	IC.AB1 ISM.AB1 OPD.AB1 OPF.AB2 PR.AB1 RM.AB3' SCM.AB3 SPE.AB1 SPP.AB3 SPTO.AB3 SQA.AB2 SSM.AB1 TP.AB2
5.1.4	**Q4.** Describe the required training curriculum for software engineering, software project managers, and affected groups (e.g., systems engineering, program management, configuration management, quality assurance, process engineering). Identify what training or orientation, as applicable, is provided for these groups in the software processes encompassing the software process areas. Describe what formal and informal vehicles are used to assure the transfer of knowledge and skills needed for individuals to perform their assigned roles effectively. **C4**	IC.AB2 IC.AB3 IC.AB4 IC.AB5 ISM.AB2 ISM.AB3 OPD.AB2 OPF.AB3 OPF.AC6 PR.AB2 PR.AB3 RM.AB4 SCM.AB4 SCM.AB5 SPE.AB2 SPE.AB3 SPP.AB4 SPTO.AB4 SPTO.AB5 SQA.AB3 SQA.AB4 SSM.AB2 SSM.AB3 TP.AB3 TP.AB4

Critical Capability	Criterion or Question	KPA.KP
5.1.4	**Q5.** Describe the measurements that are collected and analyzed to determine the status and effectiveness of the software process areas. How are they used to manage the development progress or quality of the processes or work products? How are process measurements collected by projects used at the organization level? **C5**	IC.M1 ISM.M1 OPD.M1 OPF.M1 PR.M1 RM.M1 SCM.M1 SPE.M1 SPE.M2 SPP.M1 SPTO.M1 SQA.M1 SSM.M1 TP.M1 TP.M2
5.1.4	**Q6.** How is management oversight of the software process areas achieved? Describe what periodic or event-driven reviews of the process area activities are held with the project manager and senior management. Identify any documentation describing the content of these reviews. **C6**	IC.V1 IC.V2 ISM.V1 ISM.V2 OPD.V1 OPF.V1 RM.V1 RM.V2 SCM.V1 SCM.V2 SPE.V1 SPE.V2 SPP.V1 SPP.V2 SPTO.V1 SPTO.V2 SQA.V1 SQA.V2 SSM.V1 SSM.V2 TP.V1 TP.V2

Critical Capability	Criterion or Question	KPA.KP
5.1.4	**Q7.** Describe your approach for independent review and/or audit of the software process area activities and work products. Who conducts these reviews? How often are they performed? To whom are the results reported? **C7**	IC.V3 ISM.V3 OPD.V1 PR.V1 RM.V3 SCM.V4 SPE.V3 SPP.V3 SPTO.V3 SSM.V3 TP.V3
5.2.1	**C1.** A plan for establishing and maintaining the required system and software development facilities exists, and is consistent with the program's requirements, needs, usage estimates, and schedule. **Q1 Q2 Q3 Q4**	SPP.AC14
5.2.1	**Q1.** Describe the software development facilities (host development computers, workstations, networks, memory systems, etc.) intended for the program in terms of quantity, location, availability date, capacity and response time. Describe the level of integration of the system/software development facilities (environments). **C1**	SPP.AC14
5.2.1	**Q2.** Describe the basis for determining that the facilities will satisfy the program's requirements and needs (capabilities and capacities). **C1**	SPP.AC14
5.3.1	**C1.** A program training plan exists which identifies: • The program's current and future technical, management, and skill needs • How these needed skills will be developed (informal vehicles, formal courses that need to be developed or procured from outside sources) • The resources (e.g., trainers, materials, funding, time) needed to develop these skills • The schedule for required training **Q1**	TP.AC1
5.3.1	**Q1.** How are the program's software development training needs planned and implemented? Identify the skill needs that must be addressed. What training vehicles will be used to impart those skills? What resources are planned to develop those skills? Which training vehicles are provided by the program and which are provided by the organization? Does the schedule for required training meet program need dates for skilled personnel? **C1**	TP.AC1
5.3.2	**C4.** Training records for all individuals are maintained at the organization level. A waiver procedure is applied for individuals who already possess the knowledge and skills required to perform in their designated roles. **Q4**	TP.AC5 TP.AC6

Critical Capability	Criterion or Question	KPA.KP
5.3.2	**Q4.** Describe how the organization tracks and maintains individual training records. Describe how your training process accommodates individuals who already possess the knowledge and skills required to perform in their designated roles. **C4**	TP.AC5 TP.AC6
5.3.3	**C4.** A well-defined process is followed for planning, developing, reviewing, approving, controlling, conducting, independently evaluating, and revising the organization's training program to keep it current with the organization's and projects' needs. Organizational standards exist and are followed for developing and maintaining training courses. **Q4**	TP.AC2 TP.AC3 TP.AC4 TP.V2
5.3.3	**Q4.** Describe the organizational training program. Describe the process for planning, developing, reviewing, approving, controlling, conducting, evaluating, and revising the organizational training products and process. Describe the organizational standards for developing and maintaining training courses. **C4**	TP.AC2 TP.AC3 TP.AC4 TP.V2
5.4.1	**C3.** The staff assigned to the subject program have the qualifications, technical skills, and experience in the application domains relevant to this program. **Q2 Q3 Q4 Q5 Q6 Q7 Q8**	SPE.AB4 SPTO.AB5
5.4.1	**Q8.** Describe the software management experience of your software management staff in terms of applications (domains) relevant to the subject program. **C3**	SPE.AB4 SPTO.AB5
5.6.1	**C1.** An organizational plan for improvement of system and software development process(es): • is based on action plans resulting from assessments of the system and software development processes • identifies highest priority areas for improvement • indicates resources and assignments to develop the process improvements • identifies applicable procedures • identifies how these improvements are incorporated into ongoing and future programs **Q1 Q2**	OPF.AB1 OPF.AC1 OPF.AC2
5.6.1	**C2.** The system and software process management activities of the organization are coordinated (in particular, these activities): • defining and managing changes to the organization's system and software processes • collecting and maintaining data on use of the organization's system and software processes **Q3**	ISM.AC2 OPD.AC1 OPF.AB1 OPF.AC3
5.6.1	**C3.** Senior management oversees and actively sponsors software process development and improvement activities. **Q4**	OPF.CO2 OPF.CO3

Critical Capability	Criterion or Question	KPA.KP
5.6.1	**Q1.** How is the program plan for system and software development process improvement based on action plans resulting from process assessments? Which processes are covered in a process assessment? Are system processes included? How are findings from the assessment typically addressed (e.g., through action plans which identify the changes to be made)? What are the plan's highest priority areas for improvement? What are the program's priority areas for improvement and how are these addressed in the plan? **C1**	OPF.AC1 OPF.AC2
5.6.1	**Q2.** Which activities are covered in the organizational plan for system and software development process improvement? Are group and individual responsibilities assigned and resources identified? Identify the procedures documented or referenced in your plan. How are improvements to be incorporated into ongoing and future programs? **C1**	OPF.AB1 OPF.AC2
5.6.1	**Q3.** Which individual(s) or group(s) are responsible for coordinating the system development and software development process management activities of the organization? Who is responsible for managing changes to the organization's system development and software development processes? Who is responsible for collecting and maintaining data on use of the organization's system and software development processes and making it available to other programs? How are these activities coordinated with the program? **C2**	ISM.AC2 OPD.AC1 OPF.AB1 OPF.AC3
5.6.1	**Q4.** Who oversees and sponsor the organization's activities for software process development and improvement? How are these accomplished? **C3**	OPF.CO2 OPF.CO3
5.6.2	**C2.** Systems and software development process improvement proposals are evaluated and decisions whether or not to implement them are made, based on expected benefits are relative priority. **Q2**	OPF.AC5
5.6.2	**C3.** When the decision is made to transfer a system or software development process improvement into a program, the improvement is implemented in a way that ensures: • necessary resources to implement the improvement are determined and established • the appropriate defined development process(es) and training courses are updated • consultation support is established • changes in development process performance are measured **Q3**	OPF.AC3 OPF.AC5

Critical Capability	Criterion or Question	KPA.KP
5.6.2	**C4.** Managers and technical staff are informed of the status and results of the organization's and program's activities for system and software process development and improvement. **Q4**	OPF.AB4 OPF.AC7
5.6.2	**Q2.** Describe how employee-identified and other proposed opportunities for process improvement are evaluated. What criteria are used to determine whether or not to implement a particular proposed improvement? How are benefits and priorities of proposed improvements determined? Which group(s) or individual(s) are assigned responsibility for evaluation and tracking these processes improvement proposals? **C2**	OPF.AC5
5.6.2	**Q3.** When the decision is made to transfer a system or software development process improvement into the program, what steps do you take to incorporate the improvement? What kinds of resources are assigned? How are the applicable document process(es) (e.g., program's, organization's) and training updated to incorporate the improvement? What training and consultation support do you typically plan to provide? How do you determine whether the change in process has improved technical performance and product and determine cost benefits? **C3**	OPF.AC3 OPF.AC5
5.6.2	**Q4.** What group and functions are informed of the status and results of the organization's and program's activities for system development and software development process improvement? How are they informed and how often? **C4**	OPF.AB4 OPF.AC7
5.7.2	**C1.** The S/SEE components support the program's software engineering development and management requirements, functions, methodologies, and activities. **Q1 Q2 Q3**	SPE.AC1 SPP.AC14
5.7.2	**Q3.** Describe how each tool in the S/SEE supports the software development process functions and methodologies selected for the program. **C1**	SPE.AC1 SPP.AC14

Appendix D—Commercial Off-The-Shelf Software Questions and Criteria

2	Systems Engineering		
2.7	Reuse		
2.7.4a	COTS/Reuse Software Evaluation, Selection and Management		
C1a	The contractor has a well defined process for COTS and reuse software selection that includes effective criteria to ensure that the selected products provide needed capabilities and meet system and software constraints within an acceptable level of risk. **Q1a**	**Q1a**	Describe your process for evaluating and selecting COTS and reuse software, including the criteria that each product must meet before it is considered for inclusion in a development effort. **C1a C2a**
C2a	The contractor has appropriately considered the system life cycle costs in the evaluation, selection and management of COTS and reuse software. **Q1a Q2a**	**Q2a**	What is your approach for managing COTS and reuse software on this program? **C2a C3a C4a**
C3a	The contractor has an effective plan for managing COTS and reuse software that is appropriately integrated with the software development plan and systems engineering management plan. Well defined software processes have been suitably adapted to include COTS- and reuse-specific processes, standards, and procedures. **Q2a**	**Q3a**	Describe how your software configuration management plan includes the configuration control of COTS and reuse software products selected for use on this program. **C5a**
C4a	The COTS and reuse software management plan adequately covers planning for systems engineering considerations, such as supportability, security, safety, and fault detection and isolation. **Q2a**	**Q4a**	What is your approach for ensuring the suitability of selected COTS and reuse software products for their intended purposes over the system life cycle? **C6a**
C5a	The contractor's software configuration management plan adequately incorporates processes for installing COTS and reuse software on multiple hardware platforms, managing the configuration of multiple baselines, and controlling the licensing of COTS and reuse software products. **Q3a**	**Q5a**	Describe your approach to upgrading COTS and reuse software during the system development life cycle? **C7a**

2	Systems Engineering		
2.7	Reuse		
2.7.4a	COTS/Reuse Software Evaluation, Selection and Management		
C6a	Sufficient information is obtained on COTS and reuse software products through an appropriate combination of hands-on execution, prototyping, vendor classes, and special vendor support for product maintenance and modification to ensure that the selected COTS and reuse software products are suitable for the program over the system life cycle. **Q4a**	**Q6a**	In terms of life-cycle support, how will you manage COTS and reuse software that has been operationally deployed? **C8a**
C7a	An effective plan is in place for upgrading COTS and reuse software throughout the system development life cycle to ensure continued support of all COTS and reuse software products by the vendor/controlling organization. **Q5a**	**Q7a**	Describe your approach to managing risks associated with COTS and reuse software selection, management and integration. **C9a**
C8a	The contractor's COTS and reuse software management plan provides an effective method for controlling and upgrading COTS and reuse software that has been operationally deployed. **Q6a**	**Q8a**	List the risks associated with your proposed use of COTS and reuse software that have been identified to date, and describe your risk mitigation plans for each identified risk. **C10a**
C9a	The COTS and reuse software risk management approach is adequately documented and has been successfully integrated with program and software risk management. **Q7a**		
C10a	The contractor has properly identified the principal risks associated with the proposed use of COTS and reuse software and has established effective risk mitigation plans.		

2	Systems Engineering		
2.7	Reuse		
2.7.5a	COTS/Reuse Software Integration into Development		
C1a	Existing capabilities of candidate COTS and reuse software products are given appropriate consideration in software requirements trades, especially for derived software requirements. **Q1a**	**Q1a**	Describe your approach to incorporating COTS and reuse software during the requirements analysis phase of software development. **C1a**
C2a	The contractor's system and software architectures adequately support the evolution and replacement of COTS and reuse software products. **Q2a**	**Q2a**	What is your process for including COTS and reuse software in the system and software architectures? **C2a**
C3a	The contractor has used effective techniques to assess the level of effort required to develop interfacing code for COTS and reuse software products. The assessment of the code development effort includes the types of interfaces to be developed, such as wrappers, "glue" code, Application Program Interfaces (APIs), etc. **Q3a**	**Q3a**	How do you determine the level of effort and amount of development code required to interface new COTS and reuse software products with other software elements (either developed code or other COTS/reuse products)? Where in this determination do you include the assessment of the types of interfaces to be developed for the COTS and reuse software products? **C3a**
C4a	An effective process is used for integration testing that ensures the correct functioning of the integrated software system across all COTS, unmodified reuse, modified reuse and newly developed software. The integration testing process is effectively integrated with the contractor's system and software integration test plans and procedures. **Q4a**	**Q4a**	Describe your process for integration testing of COTS and reuse software (both modified and unmodified) with newly developed software. **C4a**

2 Systems Engineering 2.7 Reuse 2.7.5a COTS/Reuse Software Integration into Development			
C5a	An effective process is used for verifying software and system requirements allocated to COTS and reuse software products, including methods and criteria for regression testing of COTS and reuse products. The verification process is effectively integrated with the contractor's system and software verification plans and procedures. **Q5a**	**Q5a**	Describe your process for verifying system and software requirements that have been allocated to COTS and reuse software products. **C5a**
C6a	The contractor's system operations and maintenance procedures include methods for scheduling COTS and reuse software upgrades, and for continuous assessment of the effectiveness of existing COTS and reuse software products over the system life cycle. **Q6a**	**Q6a**	What is your approach for integrating COTS and reuse software upgrades during system operations and maintenance? **C6a**

Appendix E—Bibliography

[AFFARS] Air Force Federal Acquisition Regulation Supplement (AFFARS), AFAC2002-1122 edition.

[Barbour 2002] Barbour, Rick; Benhoff, Melanie; Gallagher, Brian; Eslinger, Suellen; Bernard, Thomas; Ming, Lisa; Rosa, Linda; and Ryan, Charlie, Standard CMMI Appraisal Method for Process Improvement (SCAMPI), Version 1.1: Method Implementation Guide for Government Source Selection and Contract Process Monitoring, CMU/SEI-2002-HB-002, September 2002.

[Brynes 1996] Byrnes, Paul and Phillips, Mike, *Software Capability Evaluation Version 3.0 Method Description*, CMU/SEI-96-TR-002, April 1996.

[AFMCPAM 1994] Department of the Air Force, Headquarters Air Force Materiel Command, *Software Development Capability Evaluation*, AFMCPAM 63-103, Vol I, pp. 6-8 (1994).

[STSC 2000] Department of the Air Force Software Technology Support Center, *Guidelines for Successful Acquisition and Management of Software-Intensive Systems*: Volume 1, Part I, Version 3.0; May 2000.

[DoD-SEIPT 2001] DoD Software Evaluation IPT, *The Requirements for Process Evaluation Methods and Their Application*, 11 April 2001 (unpublished).

[Dunaway 1996] Dunaway, Donna K. and Masters, Steve, *CMM-Based Appraisal for Internal Process Improvement (CBA-IPI): Method Description*, CMU/SEI-96-TR-007/ESC-TR-96-007, April 1996.

[Eslinger 2002] Eslinger, S., *Piloting the Level 3-Equivalent SDCE: The Evaluation Team Perspective*, Software Technology Conference 2002.

[Haddad 1998a] Haddad, R., *Revised and Augmented SDCE Model*, The Aerospace Corporation TR-98(8550)-1, 1 March 1998.

[Haddad 1998b] Haddad, R. W., *Guidelines for the Use of SDCE*, The Aerospace Corporation TR-98(8550)-2, 1 March 1998.

[SEI-MDD 2001] Members of the Assessment Method Integrated Team, Standard CMMI Appraisal Method for Process Improvement (SCAMPI), Version 1.1: Method Definition Document, CMU/SEI-2001-HB-001, December 2001.

[SEI-MIG 2002] Standard CMMI Appraisal Method for Process Improvement (SCAMPI), Version 1.1: Method Implementation Guide for Government Source Selection and Contract Process Monitoring, CMU/SEI-2002-HB-002, September 2002.

Appendix F—Acronyms and Definitions

Acronym	Definition
AB	Ability
ABL	Air-Borne Laser
AC	Activity
ACAT	Acquisition Category
ACMS	Advanced Communications Management System
ADS	Appraisal Disclosure Statement
AEHF	Advanced Extra-High Frequency
AFFARS	Air Force Federal Acquisition Regulation Supplement
AFMC	Air Force Materiel Command
AFMCPAM	Air Force Material Command Pamphlet
AFSCN	Air Force Satellite Control Network
AI	Artificial Intelligence
ASC	Aeronautical Systems Center
AT&L	Acquisition, Technology and Logistics
C	Criterion/Criteria
CBA-IPI	CMM-Based Assessment for Internal Process Improvement
CBD	Commerce Business Daily
CC	Critical Capability
CCA	Critical Capability Area
CCSC	Command and Control Sustainment Contract
CCS-C	Command and Control System – Consolidated
CDFSII	Cloud Depiction and Forecasting System II
CDR	Critical Design Review
CDRL	Contract Data Requirements List
CD-ROM	Compact Disc – Read-Only Memory
CMM	Capability Maturity Model
CMMI	Capability Maturity Model Integration
CMU	Carnegie-Mellon University
CO	Commitment
COTS	Commercial Off-The-Shelf
CPM	Contract Performance Monitoring
CrIS	Cross-track Infrared Sounder
CSCI	Computer Software Configuration Item
DAGR	Defense Advanced GPS Receiver
DMSP	Defense Meteorological Satellite Program
DNS	Distributed Network-based System
DoD	Department of Defense
DUSD(S&T)	Deputy Undersecretary of Defense for Science and Technology
EDS	Engineering, Development and Sustainment
EELV	Evolved Expendable Launch Vehicle