

AD-A258 419



1

AFIT/GSS/LSY/92D-3

DTIC
ELECTE
DEC 21 1992
S A D

THE ADAPTATION OF THE
SEI'S CAPABILITY MATURITY MODEL
TO THE AIR FORCE SOFTWARE ACQUISITION
MANAGEMENT PROCESS

THESIS

William G. Dickerhoff Jr., Captain, USAF
William J. Sommers, Captain, USAF

AFIT/GSS/LSY/92D-3

012250

92-32212



19680

Approved for public release; distribution unlimited

The views expressed in this thesis are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

Accession For	
NTIS CRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

AFIT/GSS/LSY/92D-3

THE ADAPTATION OF THE
SEI'S CAPABILITY MATURITY MODEL
TO THE AIR FORCE
SOFTWARE ACQUISITION MANAGEMENT PROCESS

THESIS

Presented to the Faculty of the School of Systems and Logistics
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Software Systems Management

William G. Dickerhoff Jr., B.S.
Captain, USAF

William J. Sommers, B.S.
Captain, USAF

December 1992

Approved for public release; distribution unlimited

Acknowledgments

First and foremost, we must recognize the efforts of and extend our most humble gratitude to Tracy Sommers. Her tireless effort (and a great deal of red ink) have indeed improved this document far beyond our wildest dreams. With her help, we have been able to produce a thesis that we are proud to put our names on. Our families have indeed been important to both of us in this seemingly ceaseless effort. Paula and Tracys' support and patience helped us survive some of the hardest work in our career. Long days, late nights, and early mornings can keep you from the people most important to you. Our thanks and love for waiting until we ring the bell. We will get re-acquainted after graduation... Guaranteed! Finally, the direction and guidance given by Dan Ferens and John Robinson were essential to completing the process. They worked very hard to keep us on track, on time, and connected to reality. Their technical guidance and support allowed us to not only complete our thesis on time, but to make it a technically viable effort. Thank you. To everyone else, there is only one thing to say. It's over!

William G. Dickerhoff Jr.

William J. Sommers

Table of Contents

	page
Acknowledgments.....	<i>ii</i>
List of Figures	<i>vii</i>
List of Tables	<i>viii</i>
Abstract.....	<i>ix</i>
I. Introduction.....	<i>1</i>
1.1 General Issue.....	<i>1</i>
1.2 Background	<i>1</i>
1.3 Problem Statement.....	<i>3</i>
1.4 Research Objectives.....	<i>3</i>
1.5 Assumptions	<i>3</i>
1.6 Scope and Limitations.....	<i>4</i>
1.7 Benefits of the Research.....	<i>4</i>
1.8 Overview	<i>4</i>
II. Methodology.....	<i>6</i>
2.1 Introduction	<i>6</i>
2.2 Research Design.....	<i>6</i>
2.3 Exploratory Phase	<i>6</i>
2.4 Development Phase.....	<i>7</i>
2.4-1 Preliminary Software Acquisition Maturity Framework.....	<i>7</i>
2.4-2 Survey Instrument Selection.....	<i>8</i>
2.4-3 Delphi Methodology.....	<i>8</i>
2.4-4 Sample Design—Selection of Experts.....	<i>10</i>
2.4-5 Instrument Development and Testing.....	<i>11</i>
2.4-6 Validity of Delphi.....	<i>12</i>
2.5 Implementation Phase.....	<i>13</i>
2.5-1 Delphi Survey.....	<i>13</i>
2.5-2 Software Acquisition Maturity Framework Refinement.....	<i>15</i>
2.6 Summary.....	<i>15</i>

	page
III. Literature Review	17
3.1 Introduction	17
3.2 Scope and Limitations	17
3.3 Process Modeling	17
3.4 Software Process Models	20
3.4-1 Model of the Air Force Software Acquisition Process	20
3.4-2 Model of the Software Development Process	24
3.5 Acquisition Process	25
3.5-1 Description of the Air Force Acquisition Process	25
3.5-2 Description of Software Acquisition Process Management	27
3.6 The SEI's Capability Maturity Model	28
3.6-1 General Description	28
3.6-2 Process Maturity Framework	29
3.6-3 Capability Maturity Model	32
3.6-4 Software Process Assessments	33
3.6-5 Software Capability Evaluation	35
3.6-6 Critique of the SEI's CMM	36
3.7 The SEI CMM's Applicability to Software Acquisition	37
3.7-1 Maturity Level 2	37
3.7-2 Maturity Level 3	43
3.7-3 Maturity Level 4	46
3.7-4 Maturity Level 5	48
3.8 Other Software Acquisition Key Process Areas	50
3.8-1 Statement of Work Preparation	50
3.8-2 Risk Management	52
3.8-3 Data Management	52
3.8-4 Software Supportability Planning	53
3.9 Summary	54
IV. Analysis	55
4.1 Introduction	55

	page
4.2 Baseline SAMF—Delphi Questionnaire Development.....	55
4.2-1 Organization Level.....	55
4.2-2 Applicable CMM Maturity Levels.....	57
4.2-3 Applicable CMM Key Process Areas.....	58
4.2-4 New Candidate Key Process Areas.....	60
4.3 Delphi Survey Implementation.....	61
4.3-1 Selection of Experts.....	61
4.3-2 Round.....	62
4.3-3 Round II.....	63
4.3-4 Delphi Methodology Results.....	63
4.4 Responses.....	65
4.4-1 Demographics.....	66
4.4-2 Applicability of Approach.....	68
4.4-3 Organization Level.....	71
4.4-4 Framework Structure—Maturity Levels.....	72
4.4-5 Key Process Areas.....	74
4.4-6 New Candidate KPAs.....	82
4.4-7 General Recommendations.....	83
4.5 The Software Acquisition Maturity Framework.....	83
4.5-1 Modifications.....	85
4.5-2 Use of SAMF.....	89
4.6 Summary.....	89
V. Conclusions.....	90
5.1 Introduction.....	90
5.2 Research Results.....	90
5.3 Recommendations for Study Replication.....	93
5.3-1 Location of Experts.....	93
5.3-2 Dropout Rate of Participants.....	93
5.4 Recommendations for Further Research.....	94
5.4-1 Validate Results.....	94
5.4-2 Expand the SAMF into a Software Acquisition Maturity Model.....	95

	page
5.4-3 Develop an Assessment Methodology and Guidance.....	95
5.4-4 Develop Process Improvement Methodologies and Guidance.....	95
5.4-5 Research the Effects of the IWSM Approach to the Framework or Model....	95
5.4-6 Develop a Training Program Template.....	96
5.4-7 Develop a Software Metrics Program Template.....	96
5.4-8 Produce a Development/Acquisition Maturity Matrix.....	97
5.4-9 Determine if the Higher Levels of the Framework Should Apply to the General Acquisition Process.....	97
5.4-10 Develop an Acquisition Process Maturity Model.....	97
5.4-11 Other Topics.....	97
5.5 Summary.....	98
Appendix A	100
Appendix B	124
Appendix C	152
Appendix D.....	177
Bibliography.....	179
Vita (William G. Dickerhoff)	182
Vita (William J. Sommers)	183

List of Figures

Figure	page
1. Likert Scale	12
2. Likert Scale With First Round Group Response.....	14
3. Influence Diagram First Extension and Model List	23
4. Acquisition Milestones and Phases	26
5. The Five Levels of Software Process Maturity.....	30
6. The CMM Structure	33
7. Expert's AFMC Product Center Experience.....	66
8. Expert's SPO Functional Discipline Experience	67
9. Expert's CMM Training.....	68
10. Group Response for CMM Maturity Levels	73
11. Group Response for CMM Level 2 (Repeatable) KPAs	75
12. Group Response for CMM Level 3 (Defined) KPAs	78
13. Group Response for CMM Level 4 (Managed) KPAs	80
14. Group Response for CMM Level 4 (Optimized) KPAs	81
15. Software Acquisition Maturity Framework	92

List of Tables

Table	page
1. Expert Selection Criteria	11
2. Key Software Acquisition Process Management Variables	22
3. Capability Maturity Model Levels	34
4. Baseline SAMF.....	56
5. Software Acquisition Maturity Framework	84

Abstract

This study develops an Air Force Software Acquisition Maturity Framework (SAMF) by adapting the Software Engineering Institute's (SEI) Capability Maturity Model (CMM) to the Air Force software acquisition process. The SAMF's purpose is to provide the Air Force Materiel Command's product centers and program offices with criterion to assess their software acquisition maturity in a similar fashion as the SEI's CMM provides companies a benchmark to measure their organization's software production maturity. The research was accomplished through a combination of information gathering techniques and data analysis. A literature search of documentation, both within and external to the Department of Defense, identified several components of the Air Force software acquisition process. A Delphi survey collected several software acquisition experts' opinions and recommendations on the research. Based on the information gathered, the authors determined the SAMF to have the identical structure as the CMM. Twenty-one software acquisition process Key Process Areas (KPA) were identified. Definition of these KPAs was based on the research data gathered. The study produced the initial framework for an Air Force Software Acquisition Maturity Model.

THE ADAPTATION OF THE SEI'S CAPABILITY MATURITY MODEL TO THE AIR FORCE SOFTWARE ACQUISITION MANAGEMENT PROCESS

I. Introduction

1.1 General Issue

The Air Force must spend its money wisely. The budget for the Department of Defense (DoD) is getting smaller while the price of weapon systems is getting larger (SCEoverview, 1991:1.0-7). Concurrently, the threats to the U.S. are in a state of flux. Third world countries are vying to fill the power gap left by the collapse of the Warsaw pact. The threat unpredictability requires the DoD to develop and maintain weapon systems that are more flexible, secure, and capable (S/WTechStrat, 1991:ES-4). In this environment, the DoD's reliance upon software intensive systems has been increasing at an astounding rate (SCEoverview, 1991:1.0-6; S/WTechStrat, 1991:ES-4, ES-5). General Randolph (former commander of Air Force Systems Command) has stated that the Air Force has had a perfect record: it has never completed a major software program on time nor within budget (Randolph, 1989). In fact, the "typical DoD software project is a year late and double the budget" (CMMtutorial, 1991:6). The Air Force software acquisition process, like the development process, must be controlled and managed properly to gain needed improvements (Humphrey, 1988:3-4). These improvements are necessary in order to maximize future Air Force investments. The Air Force must be able to assess the software acquisition capabilities of its System Program Offices (SPO). Once the program office's capability is known, it can concentrate on improving its overall process and lowering the cost of acquiring new software.

1.2 Background

In 1987, the Software Engineering Institute (SEI), on contract to the DoD, developed a software development process maturity framework[†]. This framework, along with a questionnaire, provided the government a method for assessing the software production maturity (and by extension, the capability) of its contractors. The questionnaire was intended to indicate areas of the contractor's software

[†] A Framework is a concept that provides a structure or a base of reference to shape or adjust ideas for a particular objective or purpose (Stein and Hauck, 1980).

production process that needed improvement. A second methodology was developed to guide the contractor toward process improvement. Over time, emphasis was placed on the results of the questionnaire, and little attention was given to the framework itself. Therefore, in August 1991 the SEI published its Capability Maturity Model (CMM) that placed a greater emphasis on process improvement. Clear guidance was given for the steps an organization needed to undertake to improve its process maturity (Paulk et al, 1991:vii).

The SEI used a very structured approach toward building the CMM. At first, the SEI formulated a framework for defining an organization's capabilities for producing software. From this framework, the Software Capability Evaluation (SCE) methodology was developed for assessing the organization's capabilities. After enough data was gathered, the SEI expanded on the framework and provided clear guidance for process improvement. The steps the SEI used to develop the CMM were: 1) define a framework, 2) develop a methodology for implementing the framework, and 3) provide guidance for moving from one framework level to the next.

The DoD uses the SEI's SCE methodology in many ways. Some DoD organizations, such as Air Force Material Command's (AFMC) Electronic Systems Center and the Naval Weapons Center at China Lake use the contractor's maturity level as a determinant at source selection time (SCEoverview, 1991:3.0-16). The evaluation may also be used to judge contractor process improvement as a basis for award fees (SCEoverview, 1991:3.0-10). However, few organizations, if any, use the contractor's maturity level to determine the SPO's management approach.

Not all program offices in the Air Force have the same capability for managing software acquisition. The maturity of the individual SPO may differ greatly, but little concrete knowledge is available for making this assessment. A framework similar to that found in the SEI's CMM could be used to develop a methodology for assessing the capabilities of Air Force SPOs. This framework could provide the basis for guiding process improvement throughout Air Force software procurement. Once established, the SPO's acquisition maturity levels can be combined with the contractor's development maturity levels to define a software process maturity matrix (Mead, 1991b:5). This matrix could be used to tailor the management relationship between developer and buyer. The development of a Software Acquisition Maturity Framework would be the first step in quantifying and standardizing the relationship between the Air Force and its software contractors.

1.3 Problem Statement

The purpose of this research is to develop an Air Force Software Acquisition Maturity Framework (SAMF) based on the Key Process Areas (KPA)[†] of the Air Force software acquisition process and the SEI's CMM software development maturity framework.

1.4 Research Objectives

The primary goal of this research is to develop an Air Force SAMF. To achieve this objective, four secondary objectives must be accomplished sequentially: 1) identify the Air Force software acquisition process, 2) define key process areas within the process and organize them, 3) validate the ordered list of KPAs, and 4) develop the SAMF based on the validated KPA list and the SEI's CMM framework.

The Air Force software acquisition process will be defined through the review of all applicable DoD and Air Force documents. Software acquisition process KPAs will then be identified based on this process definition and by adapting the applicable KPAs from the SEI's CMM. Next, the software acquisition process KPAs will be organized into the order in which they must be achieved to improve process maturity. The third secondary objective will be to validate the KPA list and its ordering. This will be achieved by surveying experts^{††} at the AFMC product centers. Finally, the validated list of KPAs will be grouped into maturity levels^{†††}. These maturity levels will be patterned on the SEI's CMM maturity levels which constitute the CMM framework. The final product will be a Software Acquisition Maturity Framework that is similar in structure and content to the SEI's Process Maturity Framework.

1.5 Assumptions

Throughout the research process several assumptions will be made that guide the research effort. The first assumption is how similar the underlying software development process is to the process of managing a contractor who is developing software. This assumption facilitates adapting the SEI's CMM framework to the Air Force software acquisition process. The second assumption is that there is only one Air Force software acquisition process. This assumption shall be substantiated through

[†] The SEI's CMM defines a Key Process Area as "a cluster of related activities that, when performed collectively, achieve a set of goals important for enhancing process capability" (Weber et al, 1991:A-9).

^{††} "Experts" are Air Force personnel who met predetermined acquisition training and experience criteria.

^{†††} A maturity level represents a defined state of an organization's process. For example, a CMM maturity level consists of a list of KPAs, all of which must be achieved in order to fully qualify for that maturity level (SCEoverview, 1991:2.0-10).

literature reviews and information gathered during the interviews. The researchers are aware that the systems procured at each of the five product centers is very different in nature and each has its own unique characteristics. However, it is important to note that all five product centers are governed by the same set of directives and regulations. Hence, the researchers will develop one model to apply to all product centers.

1.6 Scope and Limitations

The SEI's CMM framework covers all aspects of software development in many different application domains. This thesis effort will not cover as broad a scope as the CMM framework. The researchers will limit the scope of the research to the Air Force SPO's managing programs which require software development to fulfill system requirements[†]. This research effort will make no attempt to generalize software procurement activities conducted by other Air Force agencies, major commands, or other DoD agencies. Therefore, the software acquisition framework developed will be applicable only to Air Force SPOs.

1.7 Benefits of the Research

The SAMF resulting from this research could be the first step toward improving the Air Force's capability of acquiring software on time and within budget. This research effort will develop a framework for measuring the capabilities of Air Force software acquisition management. Continuing research could develop and validate the methodology for assessment, the guidelines for improving maturity; and the contractor's/buyer's matrix that could be used to tailor the acquisition management approach. The final product will be a well-defined, consistent software acquisition process that is moving towards greater efficiency.

1.8 Overview

This chapter has covered the general issue and specific problem to be researched. In doing so, the research objectives were identified, assumptions stated, and the scope and limitations were declared. The benefits of this research were expressed to give impetus for further research.

Chapter II discusses the methodology employed in the research process. The chapter addresses the research phases including the survey technique used for research validation. Chapter III contains an extensive literature review. The review will include a discussion of the SEI's CMM, a discussion of the advantages and disadvantages of the CMM, a description of the DoD software acquisition environ-

[†] The terms "project" and "software program" are used interchangeably throughout this document. In both cases the terms refer to the software portion of the system being procured.

ment, and several other related topics. Chapter IV explains the data gathered and the research findings. Finally, Chapter V sets forth the conclusions of the research and any further recommendations.

II. Methodology

2.1 Introduction

This chapter describes the research methodology used for implementing this study. The chapter is broken into sections corresponding to the phases in which the methodology was implemented. Please note, some areas of the discussion allude to information explained later in the chapter. As appropriate, these areas reference following sections to maintain clarity for the reader. The reader is encouraged to follow the discussion in the order presented to understand the underlying "build philosophy"[†] of the research design.

2.2 Research Design

The research design was guided by the nature of the research objectives; the essence was to adapt an exiting maturity framework of one process to another process. The research methodology was a combination of literature reviews, analysis, and structured survey techniques. As defined by Emory and Cooper, this effort was a formal, descriptive, cross-sectional study using an *ex post facto* design (Emory and Cooper, 1991:140-141). Because the researchers had no control over the variables in the data gathering process, care was taken to use a methodology which avoided researcher bias. Accordingly, the research was accomplished in three phases: the exploratory phase, the development phase, and the implementation phase.

In the exploratory phase, existing documentation and literature were reviewed concerning the Air Force software acquisition process and the SEI's CMM. The objective was to develop a research baseline from which information about the software acquisition process, software development process, and CMM could be derived. In the development phase, the CMM framework was adapted to the Air Force software acquisition process. The preliminary Air Force SAMF was based on the research baseline developed during the exploratory phase. Next, the Delphi survey instrument was developed based on the preliminary version of the SAMF. In the third and final phase, the Delphi survey was implemented, the research results analyzed, and software acquisition maturity framework refined.

2.3 Exploratory Phase

In the exploratory phase, all relevant documentation relating to the Air Force software acquisition process and the CMM was reviewed to support the development of the baseline SAMF. Implementa-

[†] "Build philosophy" means using information from the previous phase as the baseline for the next phase.

tion of this phase was an exhaustive literature review broken into three subject areas. The first subject area reviewed was material specifically relating to the Air Force software acquisition process. This included "official" literature such as DoD, Military, and Air Force standards governing the software acquisition process. Past efforts to define the process, model the process, or define a framework of the process, such as theses and dissertations, were reviewed. In addition, reviews of current literature concerning software management, acquisition, and process control theory were conducted. The second subject area reviewed was the SEI's CMM and supporting documentation. In the review, the model was analyzed to understand its objective and framework. Particular attention was paid to the KPAs defined in the CMM to ascertain their relevance to the Air Force software acquisition process. The third and final subject area reviewed was literature regarding the CMM to determine the strengths and weaknesses of the model. The assumption was, if the SAMF was based on the SEI's CMM, it too would have some of the same strengths and weaknesses. The product of this phase was a list of Air Force software acquisition process management characteristics, an understanding of how these characteristics related to the CMM, and an understanding of how the CMM framework could be adapted to the Air Force software acquisition process.

2.4 Development Phase

The objective of the development phase was twofold: first, draft a baseline version of the SAMF, and second, develop the Delphi survey instrument to elicit recommendations and opinions of selected experts within the Air Force software acquisition community.

2.4-1 Preliminary Software Acquisition Maturity Framework. Information gained in the exploratory phase was used to guide the adaptation of the CMM framework to the Air Force acquisition process. The characteristics of the KPAs defined in the CMM were compared to the acquisition process characteristics noting the similarities and differences between the two. The SAMF was then drafted by keeping the CMM KPAs (which corresponded to acquisition process characteristics in the new framework) and defining new KPAs covering the acquisition process characteristics which were not covered in the CMM. The result was a preliminary SAMF based on the information available. However, basing the SAMF on a literature search and the educated opinions of two Air Force captains did not provide sufficient validity for the framework. Therefore, opinions and recommendations from experts in the Air Force software acquisition field were sought to improve accuracy and provide validity. A Delphi survey was used to elicit expert opinions from personnel within the Air Force. The following sections give a complete discussion on the Delphi methodology, justification for its use, and its use in the research effort.

2.4-2 Survey Instrument Selection. The survey instrument selected to verify the accuracy and provide validity to the SAMF was based on a number of considerations. First, validation of the proposed framework needed to come from people who were knowledgeable of and experienced in the Air Force software acquisition process. These people are primarily located at the Air Force Materiel Command (AFMC) product centers. Second, each product center has its own specialty and its own twist on how business is performed. To truly get an understanding of the practicality and viability of the proposed framework as it applied to the "generic" Air Force SPO, a sample of experts from all product centers was required. Third, any opinions or recommendations received would be subjective in nature and possibly conflicting. The methodology chosen had to allow for this. The Delphi survey technique met all these considerations. Most important, with the Delphi, opinions of the selected experts could be summarized objectively while minimizing researcher bias (Linstone and Turoff, 1975:5). The criteria above, coupled with the fact that the Delphi method has been used for similar applications, was the basis for using the Delphi survey technique.

Alternative instruments/techniques were looked at but not pursued for various reasons. Personal interviews were not viable due to the researchers' time constraints. A committee or working group was also not a player. The geographical separation of the product centers and the limits of time available to the selected experts for travel caused this approach to be too risky. The best survey instrument for this research effort was by far the Delphi survey.

2.4-3 Delphi Methodology. The Delphi methodology was developed by the RAND Corporation for the U.S. Air Force in the early 1950's. The objective of the methodology is to obtain and refine opinions and recommendations from a group of people knowledgeable in the subject of interest (Dalkey, 1967:1). The basis of the method is to summarize and assess a group's opinion or subject knowledge using a structured information-gathering process, to provide feedback to the people who contributed, and to allow those that contributed an opportunity to change their opinion or knowledge claims (Linstone and Turoff, 1975:3). The most common method of implementing the Delphi is through multiple rounds of questionnaires. A questionnaire, an impersonal, structured, interview technique, is an excellent tool for obtaining expert opinions (Millett and Honton, 1991:48). Questionnaires can elicit opinions from many people; the data from the questionnaires is easier to analyze as opposed to a less structured approach, and the information is saved (already recorded for you). The disadvantages are low response rates due to the impersonal nature (a response rate of 75% is excellent), misleading questions (many times caused by researcher bias), and misunderstood questions (Millett and Honton, 1991:49-50).

In the classical Delphi method, experts are requested to make recommendations and give opinions pertaining to the research subject in the initial questionnaire. The recommendations and opinions are then summarized into a list of specific topics/items. Based on this list, the second round questionnaire is comprised of questions that focus on the topics/items. The second round is sent to the experts requesting their answers and comments to the questions. Then, in successive rounds, the experts are asked to re-evaluate their responses based on the additional information. The additional information is a statistical summary of the group's opinion, a summary of the respondents' comments, and additional facts, if available. Through these successive rounds, a group consensus is usually reached (Brown, 1968:3-6; Dalkey, 1967:3-4; Martino, 1975:21-23).

A variation of the classical Delphi method is to incorporate a list of events or items in the first round questionnaire. This list will be used as a starting place for the panel. This effectively makes the first round equivalent to the second round of the classical Delphi. Another variation is to include an explanation of the context/environment the panelists should assume when answering the questions. Though there is no definitive rule as to the number of rounds required, these variations have the affect of decreasing the number of rounds required to reach a consensus. Thus, if time is short and the initial list of items and the context on how the questions should be answered is provided, then two rounds of questionnaires may be sufficient (Martino, 1975:26-27).

Feedback of statistical information to the experts is a crucial step in the Delphi technique. When the statistical information summarizes the results to a question which has a range of answers, the median response and the upper and lower quartile should be provided. If the respondent's follow-up answer falls outside the upper and lower quartile, the reasoning behind this choice should be requested (Martino, 1975:22). Furthermore, Dalkey states significant improvement can be achieved when relevant facts are provided with the statistical information. Feeding back a fact may give the respondent a signal to re-evaluate his response, changing it in the direction the quartile data indicates (Dalkey et al, 1970:25).

The key characteristic of the Delphi methodology is that it mitigates some of the problems which commonly occur in the group decision making process. "Group decision making" normally implies meetings such as committees or working groups. Problems which occur in this type of group are introduction of personal biasness, reluctance to change an openly expressed opinion, pressure to change a member's opinion, skewing the group opinion by bandwagon or majority rules affect, and discussion of irrelevant information (Brown, 1968:2; Dalkey, 1967:2-3).

Knowledge of who stated an opinion within the group may induce personal biasness. Members may feel a reluctance to change opinions for the sake of not "losing face." Pressures to change a member's

opinion may be caused by a dominant member who adversely influences the group's opinion toward his own. Also, members may compromise their own opinion due to pressure from the group as a whole thus affecting the accuracy of the group's opinion. Finally, unless tightly controlled, meetings have a tendency to wander off the subject which obscures the relevant information and opinions discussed (Dalkey, 1967:2-3).

The Delphi lessens these problems through the use of "anonymity, controlled feedback, [and] statistical group responses" (Dalkey, 1967:3). Dalkey states that by keeping the identity of the expert anonymous, the problems of personal bias, member dominance, and opinion compromises are reduced. Furthermore, by statistically summarizing the members' responses, the members can objectively re-evaluate their original responses without the pressure to conform to any one opinion. Finally, through the use of structured questionnaires and controlled feedback, only relevant information is kept on the topic of discussion (Dalkey, 1967:3-4).

2.4-4 Sample Design--Selection of Experts. A key step in implementing the Delphi survey was the selection of the experts (Martino, 1975:54). For this research, the population from which candidates (those people who meet the selection criteria but have not been officially selected) were chosen was personnel working in the AFMC product center SPOs. However, the product centers are geographically separated and each has its own "twist" on how it does business; each specializing in the acquisition of a specific system type (i.e. C³I, aircraft, space vehicle, etc.). To account for this stratified population[†], an equal number of experts from each product center was targeted for selection. Furthermore, the nature of the research dictated the sample frame^{††} should be a much smaller subset of the total population. Candidates needed a depth and breadth of software acquisition management experience and an in-depth knowledge of the SEI's CMM. Due to the sample frame's shrinking size, the researchers estimated the sample to be eighteen to twenty-four experts (three-to-four experts per product center). The expert selection criteria is shown in Table 1, page 11.

A variety of methods was used to find candidates. Candidates were found by: 1) contacting the Mission Critical Computer Resources (MCCR) focal points at each AFMC product center and requesting their advise as to possible candidates, 2) performing a search of the Military Personnel Center ATLAS database, 3) contacting the SEI for a list of Air Force personnel trained in implementing the CMM, 4) reviewing the Software Professional Development Program (SPDP) course attendance

[†] A stratified population is one that is "segregated into a number of mutually exclusive sub-populations" (Emory and Cooper, 1991:266).

^{††} A sample frame is "the list of [population members] from which the sample is actually drawn" (Emory and Cooper, 1991:247).

TABLE 1. Expert Selection Criteria

SELECTION CRITERIA	JUSTIFICATION
• AFMC Product Centers - Aeronautical System Center - Electronic Systems Center - Human Systems Center - Space and Missile Center	• The focus of the research was the Air Force Software Acquisition Process from the SPO perspective. • Each product center has its own "twist" on how it does business. A stratified sample would be more representative of the Air Force software acquisition process.
• Rank/Grade - Maj/Lt Col or GS/GM-13,14 - Captains with ≥ 8 years service - AFSC of 2724, 2716, 2885, 2825, 4935, 4925 or equivalent	• Required personnel with a depth and breadth of experience in the software acquisition field. • Required mid to upper level management - Note: Martino points out selecting the most experienced or highest position manager is not necessarily good due to the lack of time they will have to provide their opinions (Martino, 1975:53).
• SPO Functional Discipline - Projects - Engineering - Test and Evaluation - Software Quality Assurance - Software Configuration Management - Contracting - Logistics	• The focus of the effort was specific to software acquisition. Therefore, personnel selected as experts for the delphi panel should have significant experience in software acquisition. • The number of years required was based on the length of an assignment (4+ years).
• SEI's CMM Knowledge - SCE/SPA training at the SEI - Executive training at the SEI - Self/in house training (i.e. reading <i>Managing the Software Process</i> by Humphrey), SPDP, or CRAC	• Since the premise of the research effort is the SEI's CMM can be adapted to construct the SAMF, it is important the panelist understand the CMM.

records, and 5) requesting peer recommendations from other Air Force Institute of Technology (AFIT) students and candidates. Once the candidates were identified, they were contacted via the telephone or electronic mail to ensure that they met the selection criteria. If they did, they were asked to be experts for the Delphi survey. The research objectives were explained and the Delphi methodology was outlined to each expert. And, as an added benefit of personal contact with the experts, the researchers hoped to increase the survey response rate and the time spent by the experts on the questionnaires.

2.4-5 Instrument Development and Testing. The Delphi questionnaires were developed based on the data gathered during the exploratory phase. As recommended by Millet, the questions focused on the subject matter and the questionnaire's format was "user friendly" (Millet and Honton, 1991:48-49). The questions were grouped into sections corresponding to the framework of the baseline SAMF. Additional sections, such as instructions and demographics, were also added. Both

"yes/no" and "degree of agreement" questions were asked. And, throughout the questionnaires, experts were requested to state their reasoning behind their answers (Millett and Honton, 1991:49-50).

A four point Likert scale, illustrated in Figure 1, was used for the questions that requested the experts'

☐ Disagree Strongly	☐ Disagree	☐ Agree	☐ Agree Strongly	☐ No Opinion
---	-----------------------------------	--------------------------------	--	--

Figure 1. Likert Scale

opinions. The Likert scale was used because of its simplicity and because the expert responses could easily be compared (Emory and Cooper, 1991:221). Four points ranging from "Disagree Strongly" to "Agree Strongly" measured the expert's attitude towards the question. The fifth point, "No Opinion," was not part of the rating scale, nor did it imply neutrality. Rather it simply showed that the expert had no opinion about the question. The "No Opinion" box was put off to the side to emphasize that it was not part of the rating scale.

The first round questionnaire was pretested for accuracy and clarity prior to being mailed. The pretests were performed by the researchers' advisors and an SPDP instructor. Once finished, the advisors and SPDP instructor were interviewed to gather critical comments about the questionnaire. Then, the questionnaire was revised based on these comments and scrubbed by the researchers for biased or misleading questions. It should be noted that all personnel who pretested the questionnaire met the expert selection criteria. For the second round questionnaire pretest, the researchers evaluated the experts' first round comments for clarity and understandability. This evaluation was sufficient because the second round questionnaire asked the identical questions as the first round questionnaire.

2.4-6 Validity of Delphi. Over the past three decades, the Delphi technique has proven to be a valid methodology in producing a consensus opinion within a group (Millett and Honton, 1991:51). Furthermore, research has shown the reliability of the methodology is very good. Martino states some research has shown "that with a panel no larger than fifteen, consisting of a cross-section of experts in the given field, it is highly unlikely that another equally expert panel will produce a radically different median" (Martino, 1975:49). However, Emory and Cooper point out that the reliability of a research method "is a necessary but not sufficient condition for validity" (Emory and Cooper, 1991:185). This is supported by Millet who states that the validity of the Delphi is very dependent on

the quality of the expert panel and the questionnaire (Millett and Honton, 1991:51). Therefore, the researchers paid close attention to these two salient areas when implementing the Delphi survey.

To achieve a quality panel, the researchers defined the expert selection criteria, explained in section 2.4.4, to be used for the selection of knowledgeable people in the software acquisition management field. By using this criteria, the researchers were reasonably confident that the Delphi panel was comprised of highly qualified experts. This confidence was further enhanced when the researchers contacted the candidate experts to ensure that they met the criteria prior to their selection. Furthermore, biasness of the expert panel was reduced by: 1) selecting a good cross-section of software acquisition managers throughout the AFMC product centers (stratified sampling), and 2) selecting both government and Air Force active duty personnel to represent the two groups' different management techniques.

To achieve a quality questionnaire, the researchers critically evaluated its internal validity during development and pretested it for accuracy and unbiasedness. The internal validity of the survey instrument was based on content validity. Content validity "is the extent to which [the survey instrument] provides adequate coverage of the topic under study" (Emory and Cooper, 1991:180). The fact that the questionnaire was based on the baseline SAMF which in turn was based on the SEI's CMM ensured that the subject matter was well covered. Furthermore, as explained in section 2.4-5, the questionnaire was pretested by experts prior to being mailed to the panel.

The history of the Delphi methodology indicates the research method is valid if care is taken to select a quality expert panel and to produce a quality questionnaire. The researchers took these factors into account when implementing the Delphi methodology, thus providing validity to the research approach.

2.5 Implementation Phase

The implementation phase consisted of three steps: 1) execute the two round Delphi survey, 2) analyze the survey and literature review results (research results), and 3) refine the baseline SAMF based on the research results.

2.5-1 Delphi Survey. Once completed, the first round questionnaires were sent to the experts via first class mail. When possible, the questionnaires were sent to the experts' home address. This was done to by-pass the Air Force base mail distribution system which would have added to the response turn-around time. The experts were given approximately five working days to answer the questionnaire. A return, first class stamped envelope was provided for the experts to send the questionnaire back. Again, the return envelope was addressed to a researcher's home address to expedite

the turn-around time. A "drop-dead" date for responses was set for approximately ten days after the requested return date. This was done to allow those experts who may be on travel or vacation to respond to the first round.

The experts' first round responses were tallied and summarized to obtain the group's response. The group's response consisted of the median response for the Likert scale questions and the majority response for the "yes/no" questions. If a question was checked "No Opinion" it was not counted as part of the group's response. Furthermore, all comments provided by the experts were recorded and analyzed.

The second round questionnaire was a duplication of the first, however, the group's response and relevant expert comments were added for each question. (The researchers dropped those comments not considered germane, i.e. of personal nature.) Once again, care was taken to make the questionnaire user friendly. For all questions, the group's first round response was clearly marked in the answer box as shown in Figure 2. Furthermore, the comments were sequentially listed in order of agreement—

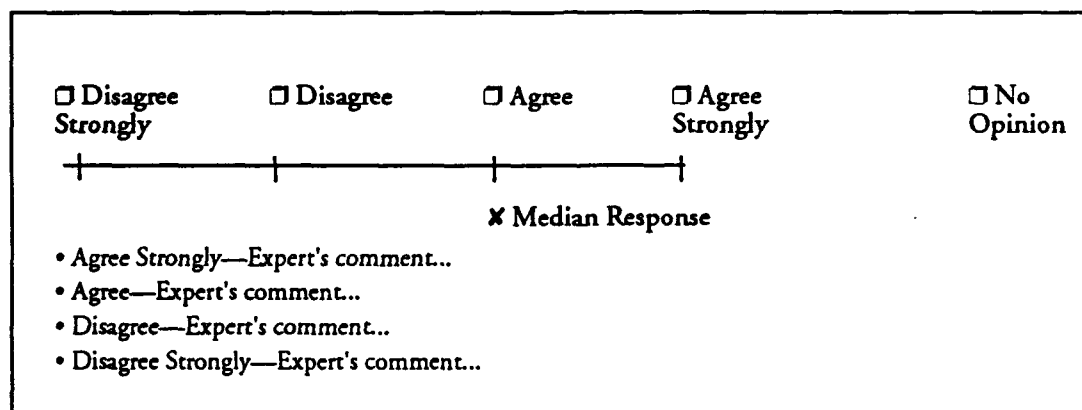


Figure 2. Likert Scale With First Round Group Response

"Yes" to "No" and "Agree Strongly" to "Disagree Strongly." Since the demographic section was dropped from the second round (this section was not needed), each questionnaire was numbered to track the questionnaire's ownership. Like the first round, the second round was sent via first class mail with a return envelope enclosed. Only those experts that responded to the first round were mailed the second. The experts were again given five working days for completion and as in the first round, the researchers set a "drop-dead" response date for ten days later. The experts were requested to reevaluate their first round answers based on the additional information provided. Each expert was also provided

a summary sheet of his first round answers for reference. The experts were instructed to leave the question blank if there was no change in their answer.

Results of the second round were compiled similar to the first. However, only responses from those experts who responded to both rounds were used in the final compilation. The additional expert comments were listed with the first round comments to allow for easy comparison between the two rounds. Once this was completed, the researchers were ready to analyze and compare the data from the Delphi survey and literature review. This data was then used to refine the baseline SAMF.

2.5-2 Software Acquisition Maturity Framework Refinement. The objective of this final phase was to refine the baseline software acquisition maturity framework based on the data gathered during the research. First, an analysis of the Delphi survey results was completed. Second, the baseline SAMF was refined based on the Delphi survey analysis and the literature review results.

Results of the two round Delphi survey were statistically summarized by calculating the median of the group's response. For each question, the comments were listed in order of agreement (shown in Figure 2) to facilitate the analysis.

Each section of the questionnaire, which corresponded to the SAMF's framework, was reviewed evaluating the expert group's opinion and comments given. Both critical and supportive comments were weighed equally and the results summarized.

The baseline SAMF was then refined based on the results of the previous step and the information gained from the literature review. Each section of the SAMF was reviewed and changes were made with the corresponding reasoning. The results of the Delphi survey were weighed heavier than the literature review data. This was done because the experts' opinions focused on the specific research objectives while the literature reviewed covered a broader topic area. Though the literature review data was valuable in guiding the researchers' decisions, the subject areas (i.e. SEI's CMM, acquisition management, process modeling, etc.) were too general when compared to the opinions of the experts. Finally, the strengths and weaknesses of this research were analyzed.

2.6 Summary

The objective of this thesis was to define an Air Force Software Acquisition Maturity Framework, similar to the SEI's CMM. This was accomplished using a three phase research methodology. The exploratory phase reviewed work previously accomplished in the research subject area, including past efforts to model the software acquisition process and the SEI's CMM. During the development phase, the researchers built the baseline SAMF and the Delphi survey instrument. Finally, in the implemen-

tation phase the researchers executed the Delphi survey and analyzed all data gathered during the research. The result of these efforts was a SAMF based on the expert opinions of Air Force software acquisition professionals and the facts gained from published material.

III. Literature Review

3.1 Introduction

The purpose of the literature review is threefold. First, publications and theses covering software process modeling or topics closely related are summarized. Second, the SEI's CMM is discussed. Finally, other works about the Air Force software acquisition process are discussed.

3.2 Scope and Limitations

The scope of the following sections differs. The first section defines process modeling. Process modeling objectives and its underlying concepts are discussed. In the next section, previous work performed in the research area is summarized. Research covering software acquisition and software development process models is discussed. The following section covers the government acquisition, software acquisition, and software acquisition management topic areas. The information contained in these sections defined the research knowledge base necessary for executing this research.

The fourth section is an in-depth coverage of the SEI's CMM. The model is summarized to the KPA level and the process used for its development is described. Next, articles from software and computer technology journals, which critique the CMM and the SEI's assessment methodology, are evaluated to identify CMM and assessment methodology strengths and weaknesses. This portion of the literature review describes the CMM, how it was developed, and how it is employed.

The fifth section relates the SEI's CMM to the Air Force software acquisition process. Each KPA's applicability to the Air Force process is addressed. Information for this critique is derived from the research knowledge base previously defined. The final section defines other software acquisition topics not addressed by the CMM. The results from this chapter are combined developing a baseline SAMF.

3.3 Process Modeling

The first questions to be answered are: what is process modeling and why apply it to software acquisition? Understanding the answers to these two questions is necessary prior to developing a process model. The following section will describe the primitive basics of process modeling and its relationship to process improvement.

Although the two questions are very general in nature, some background information must be covered. First, a definition of a "process" is needed. The following is offered by the International Standards Organization (ISO):

Process: a set of activities and tasks and their interrelationships that together transform a set of inputs into a desired output. In addition to the data upon

which it operates, a process includes methods and tools, procedures, people and skills. (Dorling, 1992:5)

In Air Force software acquisition, the inputs are in the form of a Mission Need Statement and the output is in the form of an operational system (AFR57-1, 1992:45). The tasks and activities to be performed are outlined in Department of Defense Directive (DoDD) 5000.1 and Department of Defense Instruction (DoDI) 5000.2 (5000.1, 1991:3). The rest is tailored to each individual program office.

A process model is a documented description of the process of interest. This type of model focuses on portions of the process that are important or tend to drive the variation (Osterweil, 1992). The process is documented to gain an understanding of the process and to allow for control and improvement (Osterweil, 1992).

However, process improvement does not automatically occur from simply documenting a process. Measures must be put into place to identify the process areas that cause undesirable variation. These areas are then strengthened and the problematic influences are reduced. By improving the process (i.e. reducing the influence or effect of weak process areas), the quality of the product will improve (Dorling, 1992:5).

Improving the process implies measuring the process to determine its state at any point in time. These measurements should indicate how well the process is established in the organization and how effective the process is. This type of measurement indicates the organization's "process maturity." By maturing, a process becomes more established in the organization, thus the process improves (Dorling, 1992:5).

Process maturity implies a formalized method of measuring the organization's maturity. This methodology generally takes the shape of a process assessment. A process assessment uses a defined baseline of process maturity practices or characteristics, a tool for indicating the state-of-the-practice, and a method of scoring (Dorling, 1992:6). The baseline to be judged against is the process model of interest; the tool can be in the form of a questionnaire, an interview, an inspection, etc; and the methods of scoring can take on many forms.

The primary use of such an assessment is to provide information about a user's process and to allow the user to improve his or her own process (Dorling, 1992:7). As side benefits, process assessments can instill an atmosphere of constant improvement, optimize the use of resources (i.e. reduce waste), and reduce the risk of an undesirable outcome (Dorling, 1992:10).

In summary, process modeling defines and documents important aspects of a process, measures one's process maturity, and improves the state-of-the-practice. It is done to provide insight and to improve the likelihood of a favorable result.

Why should this practice be applied to Air Force software acquisition? Chapter I indicated that the software acquisition state-of-the-practice is very poor. There is indeed room for and interest in improvement. In fact, Air Force acquisition regulations call for establishing a "software engineering and management structure to reduce the risk of software development" (AFR57-1, 1992:49). This guidance, which implements DoDD 5000.2 and DoDI 5000.3, requires the use of metrics, statistical process control, process assessments/evaluations, and risk management to improve the state of software acquisition (AFR57-1, 1992:49-54). These activities are all tools found in the arena of process modeling. It is appropriate to model software acquisition as a process in order to implement the directives of the governing regulations.

Finally, Dr. William Curtis of the SEI describes the four pillars necessary for software excellence: Process Management, Human Resources, Technical Assets, and Customer/Supplier Relations (Curtis, 1992). The pillar government can most affect is the customer/supplier pillar. This relationship is between the government program office and the developing contractor. The two organizations must agree to accept certain responsibilities and cooperate to develop a usable product (Curtis, 1992).

The government is using the process evaluation technique on its contractors during the acquisition process. In concert, many contractors are using the process assessment technique to improve their software development process. Both efforts focus on the contractor's process. To provide the correct customer/supplier relationship, the government's process must improve along with the contractor's process. A mature program office is not assured success simply because it is mature. An immature contractor could indeed cause severe problems. Conversely, a very immature program office could spoil the acquisition even if the developer is very mature (Mead, 1991a). The Air Force must do its part to improve the customer/supplier relationship.

From this discussion, it shows that process modeling provides a capability to increase the likelihood of success. Air Force software acquisition is in need of improvement. Process modeling can be applied to improve the process as well as improve the relationship between the customer and the supplier. An Air Force Software Acquisition Maturity Model has the potential to improve the software acquisition state-of-the-practice.

3.4 Software Process Models

Software process models to be explored by this review will fall in two distinct categories. The first category models the acquisition of software systems. Works presenting models related to Air Force acquisition will be addressed in the associated subsection. The second category models the software development process. Works that cover these types of models will be addressed in the second subsection.

3.4-1 Model of the Air Force Software Acquisition Process. In their thesis titled *Management Cybernetics: An Application to the Development of a Conceptual Model of the Software Acquisition Management Discipline*, Captains Peschke and Sherrill modeled the Air Force SPO's process for managing software acquisition. The authors used cybernetic[†] principles based on work by Stafford Beer (a leading expert in management principles through cybernetics) to model the process. The three objectives of their research were:

1. Investigate the current software acquisition management process with emphasis on the elements that are creating problems in the areas of employment and control.
2. Identify and define the variables (activities) which must be included in a conceptual model of the software acquisition management process.
3. Develop a conceptual model of a viable software acquisition management process which explains the system's behavior in terms of the activities and their interrelationships. (Peschke and Sherrill, 1979:11)

The foundation of their work was Stafford Beer's Recursive System Theorem.^{††} Based on this theorem, they broke the problem (defining the software acquisition process) down into its component parts and examined different SPO management levels to define key software acquisition management activities (a.k.a. variables). The advantage of this methodology was it maintained visibility throughout the whole system, which must be maintained to achieve statistical process control^{†††} (Peschke and Sherrill, 1979:15).

The systems perspective of software acquisition management differed from the traditional functional perspective. The systems perspective views all process activities concurrently while the functional

[†] "[C]ybernetics studies the flow of information around a system, and the way in which the information is used by the system as a means of controlling itself" (Beer, 1978:254).

^{††} "Within a viable organization, there is a viable sub-organization and within that sub-organization there is another sub-organization, etc, right down to the individual worker at the lowest level of the organization" (Peschke and Sherrill, 1979:15).

^{†††} "A stable process, one with no indication of a special cause of variation, is in statistical control" (Deming, 1982:321).

perspective views process activities sequentially. Hence, the functional approach to software acquisition management splits the acquisition process into phases based on the sequential software development steps (i.e. waterfall model). Each phase is managed separately, often by different directorates within the SPO. The problem with the functional approach is managers lose visibility of the system. The managers' tendency is to worry solely about their little "piece of the pie" versus worrying about how their decisions will affect the whole system. The functional approach promotes passing-off unsolved problems to the next phase, creating a snowball effect. And, the functional approach is inherently difficult to manage as a whole. The result is a program that does not meet cost and schedule. Peschke and Sherrill recognized this and therefore, chose to model software acquisition management from the systems perspective (Peschke and Sherrill, 1979:4-8, 39-41).

To gather data on the software acquisition process, the authors performed a literature review and interviewed Air Force acquisition professionals. With the data gathered, the authors identified 28 variables that had the greatest impact on managing the process. These variables, listed in Table 2 on the following page, are comprised of the SPO's, the contractor's, the user's, and the supporter's fundamental management activities and attributes. The authors operationally defined the variables which included the variables' interrelationships. With this knowledge, a software acquisition process conceptual model was developed using influence diagramming (Peschke and Sherrill, 1979:42-43).

Influence diagramming is a technique in which process variables are connected with arrows showing the influence or effect each variable has on each other (Peschke and Sherrill, 1979:22-28). The first diagramming step is to identify the core variables which have the greatest controlling affect on the process. These core variables comprise the model list around which the conceptual model is built. In subsequent steps, the remaining variables are categorized into mutually exclusive extensions lists. The first extension list contains those variables that directly influence the model list variables. The second extension list contains those variables that directly influence the first extension list variables and so on. The model and extension lists are listed in columns, side-by-side, and the variables are connected with arrows showing their interrelationships (Peschke and Sherrill, 1979:44-73).

Peschke and Sherrill's software acquisition process conceptual model is highly complex and detailed. Figure 3, page 23 depicts the concept model's key variables and their interrelationships (*note: the figure illustrates only a portion of the entire concept model*). The authors' research indicated Risk Analysis, Timely and Complete Documentation, Verification and Validation, and Formal Reviews and Audits are the core activities which have the greatest impact on the software acquisition process. Risk analysis, the "uncertainty of management parameters expressed in terms of manpower, dollars, and schedule," is the most important variable (Peschke and Sherrill, 1979:46). All other activities feed this variable.

TABLE 2. Key Software Acquisition Process Management Variables
(Peschke and Sherrill, 1979:42-43)

Management Variables	
• User involvement in development	• Early involvement of AFLC personnel
• SPO/AFPRO expertise	• Unique support requirements definition timing
• Contractor expertise	• The amount of government-furnished software
• External influences	• The age of the software being modified (when applicable)
• Planning for reprogramming capability	• Timing of the operational requirements definition
• Computed development time	• Allowed development time
• Degree of development entropy	• Accuracy of operational requirements definition
• Computer resources required for development	• Standardization of code developed
• Criticality of software being developed	• Support software available for development
• Test/verification requirement timing	• Core size of the processor to be used
• Risk analysis	• Timing cycle
• Timely and complete documentation	• Other hardware constraints
• Verification and validation	• Difficulty factor
• Formal reviews and audits	• Requirement for transportability

Specifically, the more timely and complete the contractor's documentation, the more effort put into verification and validation, and the more detailed the formal reviews and audits the more likely the software program will be completed on time and within budget (Peschke and Sherrill, 1979:46-48). Other variables which directly influenced the program's risk are: the SPO's, Air Force Plant Representative Office's, and contractor's software acquisition experience; the user's involvement in defining the program's operational requirements; the external pressures put on the program by other organizations (i.e. Air Staff); and the development time given to the program for completion (Peschke and Sherrill, 1979:53-55).

Validation of the concept model was based on the interviews, literature reviews, and the authors' confidence in the modeling process. The interviews were the primary basis for verification. Management personnel from Air Force Logistics Command, Air Force Systems Command, and Air Force

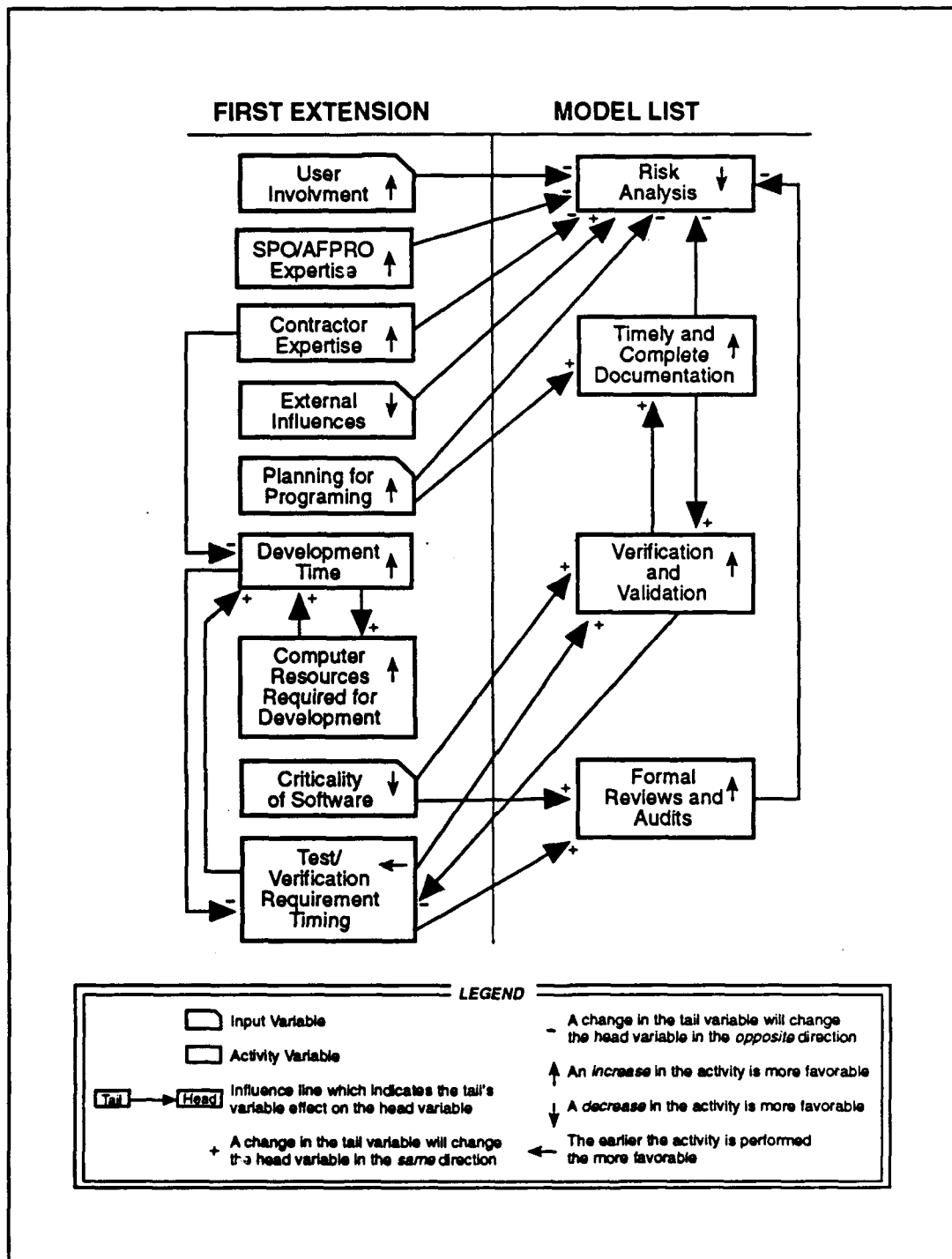


Figure 3. Influence Diagram First Extension and Model List
(Peschke and Sherrill, 1979:57)

Acquisition Logistics Division were all consulted in validating the model's variables (Peschke and Sherrill, 1979:74-76).

Summarizing their work, Peschke and Sherrill stated the concept model,

was developed in order for the manager to know what management changes to make to the software acquisition process to improve its behavior, and for that manager to gain an understanding of the complex interrelationships that exist throughout the software acquisition system. (Peschke and Sherrill, 1979:76)

3.4-2 Model of the Software Development Process. In 1991, Tarek Abdel-Hamid and Stuart Madnick modeled the software development process in their book titled *Software Project Dynamics: An Integrated Approach*. The authors developed a computer-based system dynamic model of the software development process. The underlying theme of their work asserted that little attention had been given to the software development management process (Abdel-Hamid and Madnick, 1991:6-7). On the other hand, a great deal of work has been accomplished to introduce sound engineering principles into software production since the early 1970s. Thus, the focus within the software industry has been to improve its software development technology. This lack of understanding the management process has and will continue to impede significant improvements in software development (Abdel-Hamid and Madnick, 1991:15). Accordingly, the model's objective was twofold:

1. to enhance [the managers] understanding of the software development process.
2. to make predictions about the process by which software systems are developed. (Abdel-Hamid and Madnick, 1991:6-7)

The authors' model is an integrative, system dynamic model. The model is integrative in that it looks at the software development process as a whole (not just at its phases as in the traditional modeling approach) and it integrates both the management and technical software development functions (Abdel-Hamid and Madnick, 1991:7-8). It is system dynamic in that the model uses feedback control loops which facilitates modeling "real world" activities' causes and affects on the system (Abdel-Hamid and Madnick, 1991:9-10). Unfortunately, the model is bounded to the developer's software production process. Specifically, the model is limited to the software design through test and evaluation phases. The requirements definition and software maintenance life-cycle phases are excluded. Furthermore, the authors assumed the system requirements are fixed throughout the production process. Hence, the model looks at "actions of the software development team" and does not look at actions of the external environment (e.g. the buyer) (Abdel-Hamid and Madnick, 1991:20).

Due to the model's bounding, Abdel-Hamid and Madnick's research results were not directly applicable to this research effort. However, their research methodology was similar to the methodology used in this effort and, therefore, is worth mentioning.

Abdel-Hamid and Madnick used a three-phase approach in developing and verifying the model: 1) interview software development professionals to develop a framework of the model; 2) accomplish an extensive literature review of the relevant material and perform additional interviews to fill-in knowledge gaps to refine the model; and 3) perform a case study to test their model against a software development program that had been already completed (Abdel-Hamid and Madnick, 1991:13,55-56).

The first phase consisted of gathering information through focused interviews—a semi-structured interview technique—and developing a framework of the model. For the interviews, the authors identified managers in the government and commercial industry according to predefined criteria (the authors looked for software managers with a certain minimum experience). The interviewees were notified about the general subject areas the questions would cover. Then, the interviews were conducted using a set of predefined questions. However, the questions were not asked in any particular order. Based on the interview results, Abdel-Hamid and Madnick developed a skeleton model (Abdel-Hamid and Madnick, 1991:55-57).

The second phase was an extensive literature review using the skeleton model as a research guide (Abdel-Hamid and Madnick, 1991:58). Once a literature knowledge base was established, the authors conducted an additional set of interviews to critique the skeleton model. For these interviews, Abdel-Hamid and Madnick used an unstructured interview approach (they did not use pre-defined questions). However, the questions asked were more focused in comparison to the first interview questions. Again, managers from the government and commercial industry were identified based on the same criteria as the first interview. However, the individuals interviewed in the first stage were not re-interviewed. The reasons for doing this were to increase the sample size and to decrease possible bias of the interviewees seeing the results of their initial input (Abdel-Hamid and Madnick, 1991:58-59).

3.5 Acquisition Process

The following sections cover the general research topic. First, the Air Force software acquisition environment is discussed followed by a description of software acquisition process management.

3.5-1 Description of the Air Force Acquisition Process. Air Force acquisition programs are governed by DoD Directive 5000.1 and DoD Instruction 5000.2 (5000.1, 1991:1; 5000.2, 1991:2). DoD Directive 5000.1 provides the fundamental policies for managing all defense acquisitions (5000.1, 1991:1). DoD Instruction 5000.2 is second in order of precedence (following

DoD Directive 5000.1). It "provides detailed procedures necessary to implement policies of DoD Directive 5000.1" (Cochrane, 1991:30). The following paragraphs describe the acquisition process outlined by these two documents.

These documents set forth a single DoD acquisition system, applicable to all services and all programs (5000.1, 1991:1). The major positions involved with this system are the Defense Acquisition Board (DAB), the Defense Acquisition Executive (DAE), the Service Acquisition Executives (SAE), the Program Executive Officers (PEO), and the Program Managers (PM) (5000.1, 1991:3-1 - 3-2).

Since Acquisition Category (ACAT) I[†] programs are required to follow structured acquisition phases and decision milestones, ACAT II through ACAT IV programs are also recommended to follow the same structure. As illustrated in Figure 4, the acquisition process is divided into five phases, each

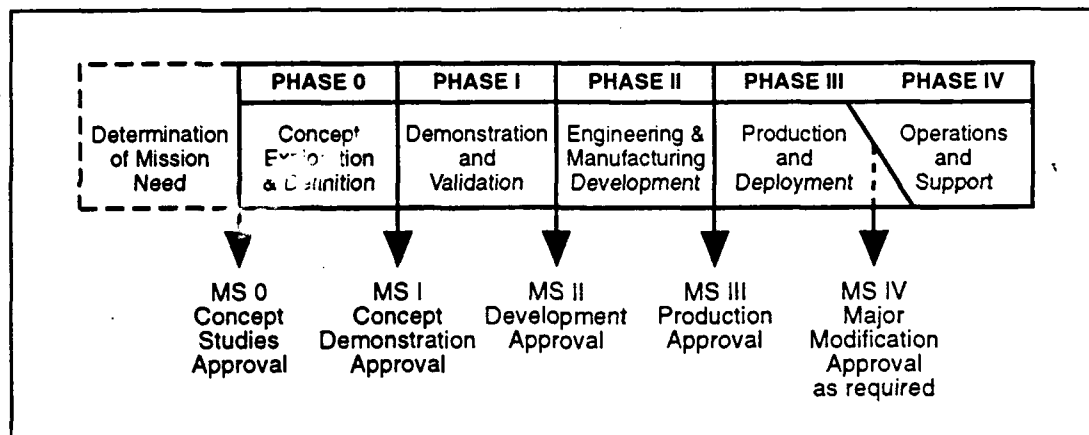


Figure 4. Acquisition Milestones and Phases (Cochrane, 1991:30)

with its own major decision point. From the information gained through a determination of mission need, a Milestone 0 decision approves the concept study. This approval begins Phase 0 of the process which is concept exploration and definition. Phase 0 ends with a Milestone I decision to approve concept demonstration. Phase I, the demonstration and validation phase, follows with its end being a Milestone II decision—development approval. Phase II, Engineering and manufacturing development, follows with its end being a Milestone III decision—production approval. After production approval, Phase III, production and deployment begins followed by operations and support. An optional Milestone IV could take place to approve major modifications (5000.1, 1991:2-1).

[†] If a program will cost at least \$200M in research development test and evaluation or \$1B in procurement, then it is categorized as an ACAT I program. ACAT I programs are also known as "major defense acquisition programs." All other programs are categorized ACAT II, ACAT III, or ACAT IV (Cochrane, 1991:32).

Before any program can pass a Milestone 0 decision, a great many activities must take place. Once a mission need is identified, the full range of options must be explored. These options begin with exploring nonmaterial changes in doctrine, tactics, or training followed by possible modifications to an existing system. The research then turns to commercially available solutions or a modification of commercial products. If none can meet the need, a cooperative research effort is attempted with an allied nation. If there is no allied interest, a joint U.S. service program is contemplated. If and only if all of these options fail will a new, single-service development program be started (5000.1, 1991:1-3).

The acquisition management for a new program must provide the proper environment to "effectively translate operational needs into stable, affordable acquisition programs" (5000.1, 1991:2-12). In order to meet this constraint, the directive suggests a great deal of planning in the areas of cost trade-offs, technology risk identification and control, cost driver identification, dual sourcing of critical or risky items, evolutionary or incremental procedures, and reliability and logistic supportability (5000.1, 1991:1-4 - 1-6). On all major defense acquisition programs operational effectiveness and suitability generate enough concern for an assessment by an independent test organization (5000.1, 1991:3-2; 5000.2, 1991:8-2).

From these two documents, it can be seen that the defense acquisition process is very structured and requires a great deal of planning. Every step in the process is described in detail in DoD Directive 5000.1 and DoD Instruction 5000.2. This includes specific procedures for managing the process at the SPO functional discipline level (Cochrane, 1991:30). However, these same documents stress that every acquisition is unique and requires its own approach. Major acquisitions that require high decision authority are subjected to further controls which mandate the writing and approving of several documents. Each of these documents requires a great deal of planning and research on the part of the program office (5000.1, 1991:2-7 - 2-12).

3.5-2 Description of Software Acquisition Process Management. In their book *Software Acquisition Management, Managing the Acquisition of Custom Software Systems*, Marciniak and Reifer describe in great detail the process of managing the acquisition of software. They state the goal of the management process is "to obtain a workable system within the framework of time and resources allocated" (Marciniak and Reifer, 1990:8). This goal holds true for the Air Force's general acquisition goal. There is no difficulty in acquiring low-risk systems on time and in budget. However, the government is often acquiring state-of-the-art systems that are high-risk development efforts. Therefore, the trick is to tailor the management process to provide the correct amount of control over the development process so the system is completed on time and at cost (Marciniak and Reifer, 1990:9).

In order to tailor the management process correctly, one must understand the software development process (covered in several other works) as well as the buying organization's management activities. The bulk of the Marciniak and Reifer text is devoted to providing a description of the buyer's management process and their interactions with the seller. The main elements of the buyer's management process are given as: 1) documentation, 2) management and engineering reviews, 3) configuration management, 4) quality assurance, and 5) assessment (Marciniak and Reifer, 1990:23-33). The authors note that even the best management process cannot overcome poorly defined quality requirements (Marciniak and Reifer, 1990:31). These requirements should drive the formation of the management interrelation and should lend themselves to measurement. The indicators used to measure the software program's quality should be identified at the beginning of the effort and agreed to by both parties (Marciniak and Reifer, 1990:32).

The authors provide in-depth discussions on the management activities each organization must conduct to help ensure a quality product. These discussions include office activities, management techniques, strategies, and planning (Marciniak and Reifer, 1990:9-225). A great deal of detailed information is stated about the practice of acquiring new software products. Virtually all of this information can be applied to the Air Force acquisition process.

3.6 The SEI's Capability Maturity Model

In 1987, the SEI contracted with the DoD to develop a method for evaluating a firm's ability to produce software. As a first step to produce this methodology, the SEI constructed a Software Process Maturity Framework. This framework is fully defined in Humphrey's *Managing the Software Process*. From this framework, the SEI developed a methodology for evaluating a firm's capability to produce quality software as well as a methodology for a firm to assess its own process and develop action plans for improvement (Bollinger and McGowan, 1991:25-6; Card, 1991:102). After several years of use, the SEI saw a need to update and expand their framework and assessment/evaluation methodologies. Their result was the Capability Maturity Model (Paulk et al, 1991:viii). The following sections discuss the maturity framework, its associated methodologies for assessments and evaluations, and the CMM.

3.6-1 General Description. The CMM consists of the maturity framework, the Software Process Assessment, and the Software Capability Evaluation. The CMM expanded the original framework to include detailed guidelines for process improvement. From the CMM, a new set of questionnaires will be developed for the assessment and evaluation activities (Paulk et al, 1991:viii). The following subsections describe each of these items.

3.6-2 Process Maturity Framework. The following pages are a summary of two works written by Watts Humphrey, a research scientist at the SEI and founder of its software process program. The first work is a book titled *Managing the Software Process*, the second, an article titled "Characterizing the Software Process: A Maturity Framework," which appeared in *IEEE Software* in March of 1988.

To overcome software problems, the development process must be controlled, measured, and improved. The SEI framework provides a basis for measurement and improvement and allows management to gain control of the process. An objective of the framework is to allow management to produce products while improving the organization's capabilities. Milestones or interim goals are also provided to aid the improvement process. Finally, the framework is also designed to reflect the actual way organizations improve their software development capabilities (Humphrey, 1990:3-4; Humphrey, 1988:73-74).

The SEI characterizes an organization by placing it into one of five maturity levels, pictured in Figure 5 on the following page. These maturity levels represent five steps of process improvement. The more mature activities performed, the higher the organization's level of maturity. The process maturity framework names the five levels as: 1) the initial level, 2) the repeatable level, 3) the defined level, 4) the managed level, and 5) the optimizing level. The following paragraphs are devoted to a discussion of the five maturity levels (Humphrey, 1990:5).

The initial maturity level is often referred to as the ad hoc level and is generally characterized as chaotic. There are no formal management procedures, cost estimation, or project planning. If software development tools are in place (which they seldom are), they are not well integrated or used throughout the organization. There is very little change control; the product being developed is often a moving target. In this stage, management has very little insight into the problems or issues facing the software development team (Humphrey, 1990:5; Humphrey, 1988:6-8).

The easiest way to detect an organization at the initial maturity level is to observe it during a crisis. If the organization reverts to a code-and-fix strategy, it is likely to be a very immature organization. Any development methodology used during normal operations should be just as valid during crisis situations. A turn away from the methodology shows a lack of understanding the software development process (Humphrey, 1988:75).

To mature to the repeatable level, an organization must institute basic project controls. Project management must be setup to ensure proper control of resource commitment. Senior management must gain insight into the software development process by reviewing and approving all project plans prior to company commitment. An independent quality assurance group must be setup to ensure

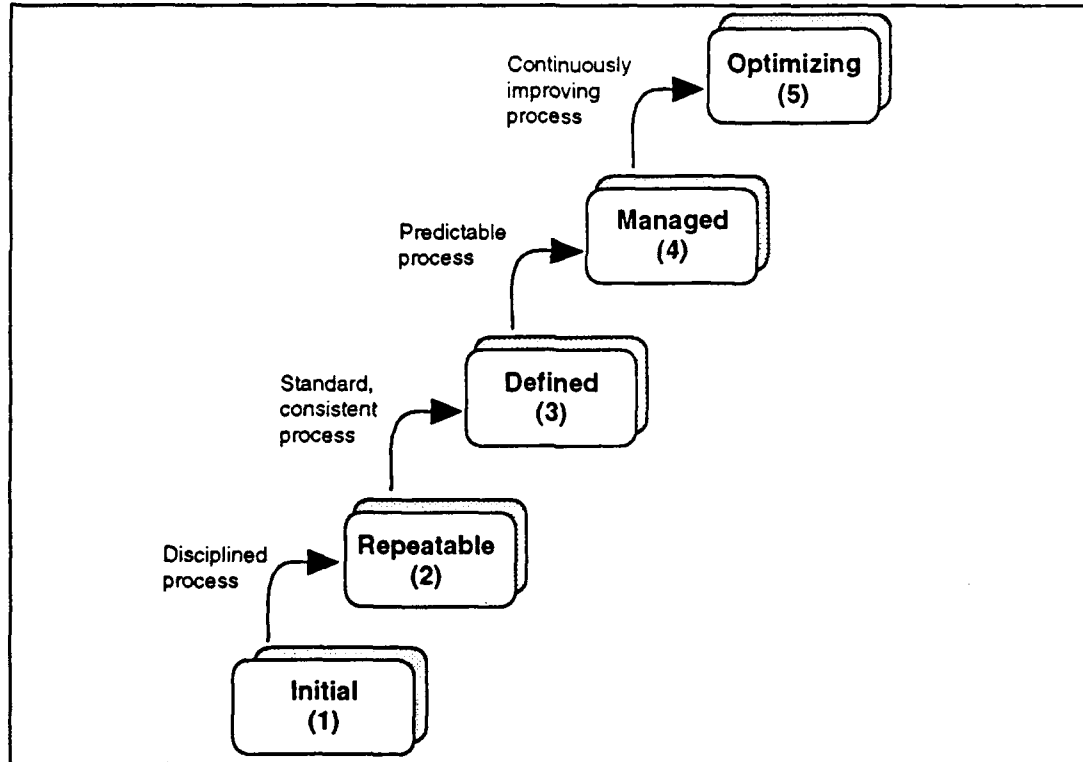


Figure 5. The Five Levels of Software Process Maturity
(Paulk et al, 1991:8)

compliance of the appropriate conventions and procedures. Finally, the organization must control the change control process (Humphrey, 1990:8-9; Humphrey, 1988:75-76).

An organization at the repeatable maturity level establishes, plans, and controls its commitments. Its strength lies in its historical knowledge and experience. From this, the organization can make and meet schedule and cost predictions. Organizations at this level tend to do well at projects they understand and have experience in. The danger lies in uncontrolled introduction of new technology and attempts to solve completely new problems. The knowledge base used to predict the time and cost estimates may no longer be valid; the project plans may not fit the new development. Care must be taken because risk may increase when something new is introduced (Humphrey, 1990:8-9; Humphrey, 1988:75-76).

The steps for maturing beyond the repeatable level are to refine the software development process. The first action needed is to establish a software process group. The main function of this group is to establish and improve the software process. A software development architecture (or life cycle) must be adopted to define the technical and managerial activities that take place during the development.

Finally, structured software engineering practices must be introduced, if they are not already present (Humphrey, 1990:9-10; Humphrey, 1988:76-77).

The next maturity level is the defined level. At this level, the organization has the ability to examine and improve its own process. Sound software development practices are established and followed. The organization has achieved the foundation for major improvement (Humphrey, 1990:11).

Again, several steps must be taken to mature beyond this level. Management must define a set of basic measurements to begin collecting data. The data should quantify the benefits of the different process stages and point out areas needing improvement. An organization-wide database should be setup to hold the data being taken. Resources must be committed to obtain and maintain this data store. Each product produced must be assessed and management informed of its quality. Actions can then be taken to correct deficiencies in the process (Humphrey, 1990:10-11; Humphrey, 1988:77-78).

If all the actions listed above are taken, the organization has matured to the managed level. At this level, the organization collects process data and makes changes to the process based on the data collected. Management must focus on improving the process to achieve better quality results. A major drawback of this maturity level is the cost to gather the data. Data is not cheap, so management must carefully choose which data is to be collected. It is important to note that this data should be used to judge only the process, not to compare different divisions. The data collected for process improvement shows nothing about a project's complexity or the requirement's solidity. The data taken should be solely for that organization's process evaluation (Humphrey, 1990:10-11; Humphrey, 1988:77-78).

To achieve the highest maturity level, only a few steps are left to implement. Management must support and implement automatic data collection as well as analyze and enhance the process. Automatic data collection reduces the error inherent to human collection as well as reduces bias. Management would be wasting time and money if it did not use the data it was collecting. Therefore, management must also plan for the evaluation of the data. The data must be used to continually optimize the development process (Humphrey, 1990:12; Humphrey, 1988:78).

The optimizing level allows talented people to become more productive. On the other hand, management can tell where the talent is needed and provide them the proper support. An optimizing process provides a professional environment that allows qualified personnel to work productively (Humphrey, 1990:12).

As organizations mature from level to level, their capability to produce quality software increases. There is no information on how long it takes for an organization to progress from the initial level to the optimizing level. However, the key is management commitment. Management must take an active

interest in the process and be willing to commit resources (money, people, and time) toward achieving the goal (Humphrey, 1988:78-79).

3.6-3 Capability Maturity Model. "The CMM is a framework representing a path of improvements recommended for software organizations" (Paulk et al, 1991:23). The CMM took the structure of the maturity framework and expanded it into a conceptual model. Four primary uses of the CMM are:

- Assessment teams will use the CMM to identify improvements needed in the organization.
- Evaluation teams will use the CMM to identify the risks of selecting among different contractors for awarding business and as a tool to monitor contracts.
- Managers will use the CMM to understand the activities necessary to implement an improvement program across their organization.
- Process improvement groups will use the CMM as a guide to help them define and improve the software process in their organization. (Paulk et al, 1991:23)

Because of its various uses, the CMM is written as a layered product. Senior managers can understand the process improvement concepts by reading only the high-level discussions while project managers can obtain detailed process improvement methodologies at the lower levels (Paulk et al, 1991:23). The following paragraphs describe the different CMM maturity levels as described in *Capability Maturity Model For Software* written by Paulk and others.

The CMM's basic structure is the process maturity framework described earlier in this chapter. However, each maturity level is then broken out into several KPAs that must occur in order for an organization to reach that maturity level. Each KPA is further broken down into key practices that are conducted to achieve the goals of the KPA (Paulk et al, 1991:23). This structure is depicted in Figure 6 on the following page.

As stated, the maturity levels used in the CMM are equivalent to those in the process maturity framework. KPAs are used to "identify the issues that must be addressed in order to achieve a maturity level" (Paulk et al, 1991:24). From the previous discussion of the framework, it is understandable that the KPAs are derived from the required activities. In order to achieve process improvement, KPAs that are grouped within a maturity level must be accomplished concurrently. Once an organization matures past a certain level, they cannot disregard the lower level KPAs. These areas are still relevant to the process. The emphasis can now be shifted from the initial development to key processes improvement (Paulk et al, 1991:23-27).

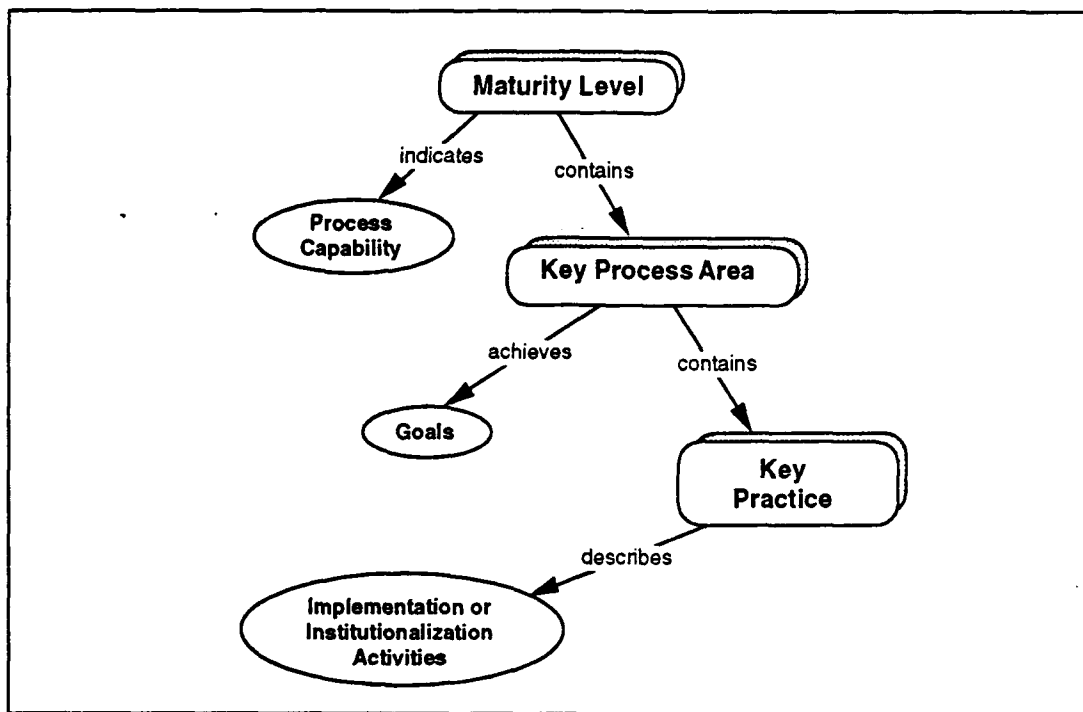


Figure 6. The CMM Structure (Paulk et al, 1991:25)

Since there are no prerequisites for a level 1 organization, there are no KPAs. The remaining four maturity levels are shown in Table 3, page 34 with their corresponding KPAs.

Each of these KPAs is further decomposed into Key Practices. Key Practices are the policies, procedures, and activities that are instrumental in implementing its KPA (Paulk et al, 1991:29). Each Key Practice has its own set of Key Indicators which indicate if the KPA goals are being satisfied (Paulk et al, 1991:32). In order to provide a greater depth of direction, the CMM also provides for each KPA: 1) a set of goals, 2) the commitment needed to perform, 3) the ability needed to perform, 4) the activities needed to perform, 5) the monitoring of the implementation, and 6) the verifying of the implementation (Weber et al, 1991:O-21).

The individual can use the CMM to any level desired. The model layering makes it easy to use for both senior management and project managers. It offers information ranging from general direction to detailed guidance. The model provides a great deal of information for anyone interested in software process improvement.

3.6-4 Software Process Assessments. In developing the framework for DoD, the SEI also provided a tool that contractors could use to improve their process and not involve the

TABLE 3. Capability Maturity Model Levels
(Paulk et al, 1991:28)

Maturity Levels	Key Process Areas
Level 2	• Requirements Management • Software Project Planning • Software Project Tracking and Oversight • Software Subcontract Management • Software Quality Assurance • Software Configuration Management
Level 3	• Organization Process Focus • Organization Process Definition • Training Program • Integrated Software Management • Software Product Engineering • Intergroup Coordination • Peer Reviews
Level 4	• Process Measurement and Analysis • Quality Management
Level 5	• Defect Prevention • Technology Innovation • Process Change Management

government. With it, a contractor could assess its own maturity level and institute steps toward process improvement. An important distinction of the process assessment is that the results are completely confidential. The software process assessment provides a methodology for firms to determine their maturity level as defined by the maturity framework, to identify possible areas for improvement, to educate senior management of the problems, and to provide guidance for planning process improvement (Humphrey et al, 1991:14). The following paragraphs summarize the procedures used in a software process assessment as they are stated in the *IEEE Software* (July 1991) article by Watts Humphrey, Terry Snyder, and Ronald Willis titled "Software Process Improvement at Hughes Aircraft."

The first step of the assessment process requires senior level management support for process improvement; the firm must contract with the SEI for an evaluation to be done. Management must provide funding for the assessment. Since management is willing to fund the assessment, they should be more willing to act upon the recommendations provided (Humphrey et al, 1991:12,14).

The next step is to select five or six projects that the firm is currently working on to serve as a representative sample of the firm's development process. The project managers are then given a questionnaire to answer. The questionnaire covers all areas of each project's development process. Once

the questionnaires are completed, the evaluation team begins a four day on site assessment (Humphrey et al, 1991:12,14).

The on site assessment begins with a briefing to management covering the ground rules and schedule; management cooperation and support is also sought. The assessment team then meets privately to review the questionnaires' answers. The team interviews several individuals to clarify the questionnaire responses. If necessary, project leaders are also interviewed to gain further insight into the development process. These activities generally take the entire first day (Humphrey et al, 1991:12,14-15).

The second day consists of discussions with technical individuals to gain further insight into the exact nature of the different aspects of the process. Day three is spent reviewing relevant documentation and re-interviewing individuals as needed. At this point, the team has completed the research portion of the assessment. All that remains is to present the findings to the firm's management on day four (Humphrey et al, 1991:12,15).

The findings are presented to the firm in two forms: a briefing is given to senior management during day four, and a final report is written. The findings highlight the highest priority areas for process improvement. After the briefing and final report, the assessment team produces an action plan to address the needed process improvements. Upon request, the SEI will review and comment upon the action plan (Humphrey et al, 1991:15).

3.6-5 Software Capability Evaluation. If one would liken the software process assessment to hiring a tax consultant, a software capability evaluation would be like an audit by the internal revenue service (Bollinger and McGowan, 1991:26). A firm uses the assessment methodology to improve its own software development process while external organizations (i.e. the DoD) use the evaluation methodology to determine a firm's strengths and weaknesses. As stated earlier, the DoD uses the software capability evaluation for input at a source selection as well as for awarding incentive fees. Therefore, the government uses the evaluation to gain insight into the firm's software development process capabilities (SCEoverview, 1991:2.0-13).

The evaluation begins when a government agency determines that a firm's software capabilities need to be evaluated. The government either sends a team to the SEI for training or hires an approved evaluation team. The team sends a questionnaire to the firm's management (the same questionnaire used in the software process assessment). The team then conducts a three day, on site evaluation during which they interview individuals to clarify answers or to gain further insight into the process. All of the

firm's positive answers must be substantiated by documentation. Accordingly, the team spends a great deal of time reviewing the firm's documents and plans.

At the end of the third day, the team produces an evaluation report that assigns a maturity level to the firm (a number from one to five corresponding to one of the five maturity levels) and identifies the firm's strengths and weaknesses. The results of the evaluation are the property of the government agency and can be used in any manner it desires. The results are not necessarily released to the firm, but can be (SCEtraining, 1991:2.3-1 to 2.3-15).

The software capability evaluation is used to gauge the strengths and weaknesses of a given contractor's software development process. There is no attempt made to provide information on improving the process. This differs from the assessment methodology which focuses upon providing guidance for improving the firm's process. Indeed, the evaluation serves as an audit while the assessment serves as a consultation.

3.6-6 Critique of the SEI's CMM. Since the SEI's CMM has recently been released, much of the criticism found in the literature is of the pre-CMM Process Maturity Model. The following paragraphs will cover the criticisms and, where possible, detail how the CMM has addressed those areas. This summary should give the reader an idea of why the CMM was produced and the direction the SEI is headed.

As stated earlier, the CMM defines a synthetic benchmark (Card, 1991:102). No "ideal software development organizations" were available to base the model on. The model is based on statistical process control principles as presented by Deming and Juran with additional information gained from several IBM projects (Humphrey, 1990:xi-xii). One might ask: Is the model valid? The CMM does not expand upon the validation of the model. However, in developing the CMM, the SEI used an extensive amount of information gathered through the many process assessments and capability evaluations in addition to a great deal of feedback from industry and the government (Paulk et al, 1991:vii).

Secondly, level one has no meaning other than failure; the only qualifications for being a level one organization is not meeting the criterion for level two (Bollinger and McGowan, 1991:30). By this definition, there can be a great variety of organizations represented as level one. This is shown by the fact that as of 1991, more than 80% of the organizations that answered the SEI questionnaires were assigned to level one (Card, 1991:102). In fact, an organization that can legitimately answer yes to all but two questions will still be categorized as a level one organization (Bollinger and McGowan, 1991:32). The CMM does not address nor clarify this point.

The original questionnaire provided a maturity level based on the answers to some 85 yes or no questions. A great deal has been written about the statistical inaccuracies associated with this sparse-data analysis technique (Bollinger and McGowan, 1991:31-32). The CMM addressed this point at a very fundamental level by introducing a structure that allows more data points to be obtained. The CMM divides the maturity levels into 18 KPAs; each has its own key practices (343 in all) that can be verified through key indicators (Baumert, 1991:78). The new SEI questionnaire (which has yet to be released) will be based upon these key indicators (Baumert, 1991:78). The incorporation of key practices and key indicators gives the CMM a much broader statistical basis in assigning maturity levels.

Other areas not specifically addressed by the CMM are the effects of automation, the misconception that maturity equals capability, and the possibility of backward motion. The model does not distinguish between the capabilities of an organization using a fourth generation language and one using machine code (Bollinger and McGowan, 1991:30). The SEI does not make a definite distinction between an organization's maturity and its capability. One could define capability as "the quantitative level of performance...in terms of variables like error rate and cycle time" (Card, 1991:103). The assignment of a maturity level does not provide assurance that the organization's capability is compliant with this definition (Card, 1991:103). The SEI also does not address the possibility of an organization digressing down the maturity scale (Topper and Forgensen, 1991:9). While this seems possible, the CMM offers no discussion of this phenomenon.

Initially, some government agencies are using an organization's maturity level as a determinant in source selection (Card, 1991:102; Bollinger and McGowan, 1991:28). This type of use places a great deal of emphasis on an organization's "score." It is indeed likely that an organization will have strengths in certain areas and weaknesses in others (Topper and Forgensen, 1991:9). Taking into account the scoring problems as listed above, reliance upon a single number score may not be wise. To de-emphasize this effect, Baumert suggests that the product of an evaluation, based upon the CMM, should be a Key Process Profile (Baumert, 1991:78). This method will give the decision makers a great deal more information than a single digit.

3.7 The SEI CMM's Applicability to Software Acquisition

The following sections describe in detail how the CMM's KPAs apply to software Acquisition in general and specifically, to Air Force acquisition. The KPAs are covered in the order that they appear in the CMM—by maturity level.

3.7-1 Maturity Level 2. The KPAs for level two are: requirements management, software project planning, software project tracking and oversight, software subcontract management,

software quality assurance, and software configuration management. The following paragraphs address the relevance of these topics to acquisition.

3.7-1.1 Requirements Management. The Statement of Work (SOW) and system specification set forth the requirements at the time the program is started. However, new requirements are destined to be introduced (Marciniak and Reifer, 1990:78). The reasons for the new requirements vary from changes in the operational concepts to changes in technology (Marciniak and Reifer, 1990:78). If the requirements are not addressed in the system, the software could be useless when it's delivered from the developer. Therefore, "requirements must be managed continuously because their impact on development is great" (Marciniak and Reifer, 1990:78).

The current DoD software development standard used by the SPO requires the contractor to document the traceability of requirements to each Computer Software Configuration Item (CSCI), Computer Software Component (CSC), and Computer Software Unit (CSU) (2167A, 1988:14). The contractor must also document that all requirements in the Interface Requirements Specifications (IRS) and Software Requirements Specifications (SRS) are tested by cases identified in the software test description (2167A, 1988:16). By inference, the DoD requires the contractor to manage these requirements.

In order to adequately manage requirements, the contractor should have a well-defined process (Marciniak and Reifer, 1990:79). The requirements management process must be completely integrated with the software development process. It should also be closely linked to the configuration management activities. Furthermore, the buyer and the seller must agree to the process and both must play active roles for it to be successful (Marciniak and Reifer, 1990:79-80).

Some possible evaluation criteria the buyer can use to judge the contractor's ability to manage requirements could be:

1. Does the seller understand the importance of managing requirements?
2. Does the seller have hands-on experience with managing requirements on similar projects?
3. Does the seller have management mechanisms and procedures for requirements management?
4. Does the seller's procedure provide for an authoritative customer role in the requirements management process? (Marciniak and Reifer, 1990:79-80)

By assuring that the contractor can answer yes to the questions above, the program office can gain confidence that the contractor can operate in an environment where change is inevitable.

3.7-1.2 Software Project Planning. In its policy documents for acquisition management, the federal government recognizes that effective planning is essential for success (5000.1, 1991:1-4). Marciniak and Reifer also note that planning for the management of an acquisition program is very important and should not be ignored (Marciniak and Reifer, 1990:181). Humphrey further states that a planning process should be written to provide guidance for all such programs (Humphrey, 1990:87). This written policy should directly address the procedures for establishing the program goals and objectives, for writing the contractual work breakdown structure, and for estimating the project costs, schedules, and resources (Humphrey, 1990:87).

In rewriting the acquisition guidance (i.e. DoD Directive 5000.1 and DoD Instruction 5000.2), the DoD attempted to produce a document that could serve as a written policy for managing an acquisition program. The intent was to "establish a disciplined, rigorous acquisition management process with clear uniform standards" (5000.1, 1991:1-9). Government guidance must be broad in scope in order to cover the many types of acquisition programs attempted. Therefore, these documents stress the need for a well thought-out program plan and acquisition strategy that is tailored to meet the objectives of the particular program (5000.2, 1991:5-1). DoDI 5000.1 and DoDD 5000.2 provide a frame or reference for tailoring the acquisition process (5000.2, 1991:5-1).

Specifically, the program management's software acquisition approach, plans, and decisions are to be documented in the Computer Resources Life Cycle Management Plan (CRLCMP) (5000.2, 1991:6-D-2). This plan identifies that all major computer/software related risk areas include support and safety issues. The CRLCMP is also directed to cover such topics as the program objectives, risk areas, costs, development methodologies, and evaluation criteria (5000.2, 1991:6-D-2). To insure that the software is developed with logistic supportability in mind, the CRLCMP is developed in conjunction with the integrated logistics support plan (5000.2, 1991:6-D-3).

Planning for the program's acquisition management is indeed very important. The DoD directs program offices to plan for the process and provides guidance for producing a tailored acquisition strategy and program management plan. Furthermore, the DoD sets aside a separate document for documenting the management approach and decisions for the program's software.

3.7-1.3 Software Project Tracking and Oversight. This KPA encompasses the review of project accomplishments and the comparison of the accomplishments against documented plans (Weber et al, 1991:L2-29). The previous section discussed the planning process and its importance to the program office. In conjunction with this, the contractor should produce a Software Development Plan (SDP) detailing the specifics of the software development effort. DoD-STD-2167A

requires the contractor to conduct the software development activities following the procedures outlined in the SDP (2167A, 1988:9). Furthermore, the DoD Directive 5000.2 states that the program office must establish a control system to measure the contractor's performance (5000.2, 1991:6-A-4, 6-D-4). Hence by regulations, the Air Force program offices must conduct project tracking and oversight.

Is this process really necessary though? Humphrey states that sound management requires a knowledge of the project status (Humphrey, 1990:103). Marciniak and Reifer warn against relying on oversimplified reporting or broad interpretation of the data (Marciniak and Reifer, 1990:133). This point directs the manager to gather the data and produce his own interpretations and reports. In effect, track the software program and ensure that oversight is obtained.

This process is a two-way street, however. The contractor must be willing to provide the government with the data and mechanisms for proper management oversight. The program office must also have the capability to perform the necessary actions to interpret the data and have an interest in taking advantage of the oversight provided by the contractor. Both parties must have the capability and desire to provide constant surveillance over the program (Marciniak and Reifer, 1990:95). This constant attention is necessary because software programs tend to "get into trouble a little at a time" (Marciniak and Reifer, 1990:133).

A discussion of software project tracking cannot be complete without mentioning management indicators and metrics. The purpose of the indicators and metrics is to gain knowledge into the progress of the project (Marciniak and Reifer, 1990:98). Many different indicators and metrics exist and many different uses are employed for them. An entire thesis effort could be undertaken in this area alone. It is important that these tools are used toward some goal. There are many ways to select the proper set of tools to be used on a project. However, management indicators and metrics require a great deal of knowledge and experience to be used properly (Marciniak and Reifer, 1990:98). A metric program, therefore, must be well thought-out and disciplined.

From this section, one can see that project tracking and oversight is indeed helpful to acquisition management. Furthermore, it is required that the Air Force program offices perform these functions.

3.7-1.4 Software Subcontract Management. According to the SEI's CMM, subcontract management entails selecting qualified contractors; contracting the proper requirements, processes, and requirements; ensuring the commitments are agreed to by both the prime and subcontractors; and tracking the subcontractor's actual results against the planned commitments. In order to perform these functions, the prime should have an established written policy to guide the

process. Also, a single individual should be given the responsibility to manage the interface between the prime and the subcontractor (Weber et al, 1991:L2-45 through L2-46).

Humphrey provides the processes an organization attains as it matures in regard to subcontract management. Early processes included project estimating and tracking, and establishing the functions for reviewing the subcontractor's Configuration Management (CM) and Software Quality Assurance (SQA) functions. Later, formal subcontractor selection procedures were established and documented. In an advanced maturity level, means are established to connect the prime and subcontractors' development environments. And a standard set of management indicators and metrics is selected and applied to all projects (Humphrey, 1989:448-453).

The activities listed above are indeed important to the software development process. However, from an acquisition management perspective, are they activities the SPO should perform? No, not directly; it is the prime contractor's function to perform the subcontractor management activities and the SPO's function to contract for their accomplishment. Hence, the important software acquisition management aspect of this KPA is that the requirements for proper subcontract management are stipulated in the prime contract.

3.7-1.5 Software Quality Assurance. SQA is responsible for giving senior management the confidence they need to produce software products and process activities that conform to the standards and applicable specifications (Weber et al, 1991:L2-63; IEEE, 1983 as stated in Marciniak and Reifer, 1990:93; Humphrey, 1990:141). The SQA organization is employed to ensure independently that the appropriate standards and procedures are followed and that any non-compliance issues are reported to senior management (Weber et al, 1991:L2-61).

However, the Air Force program office's SQA function is not to perform the SQA activities but to confirm that the contractor has indeed done these activities. The program office must ensure that the developer performs as required by the contract. It is the buyer's job to tell the seller what to do, not how to do it, nor to do it for him (Marciniak and Reifer, 1990:71). Therefore, the role of an SQA function in an Air Force SPO differs from that discussed in the CMM.

The DoD acquisition guidance requires that the SQA standards be contracted (5000.2, 1991:6-D-3). The quality standard (DoD-STD-2168) states as its purpose:

The objective of the contractor's software quality program shall be to assure the quality of (a) the deliverable software and its documentation, (b) the process used to produce deliverable software, and (c) non-deliverable software.
(2168:2)

The standard also requires the contractor to plan adequately, fund, and provide resources for conducting this quality program (2168:2). It is the SPO's SQA function to ensure that the SQA plan is adequate for the task and that the contractor implements the plan as documented.

Ensuring that the quality plan is employed properly is a difficult task. The program managers must first completely understand the quality requirements. Secondly, they must understand how the contractor's plan addresses these requirements. Finally, the contractor and the program office's SQA functions must work well together to ensure their goals are accomplished. This process requires a great deal of planning and communication (Marciniak and Reifer, 1990:184).

The government's quality assurance function is very important. The SPO must ensure SQA is required by contract and properly planned for and implemented by the contractor. However, the distinction between the SPO's system quality assurance and its software quality assurance is not significant enough to warrant an SQA KPA. As discussed above, implementing SQA is the contractor's responsibility. The SPO's SQA activities are part of the overall system quality assurance effort. Software specific quality assurance contracting and monitoring activities can be incorporated into other KPAs such as requirements management and software project tracking and oversight.

3.7-1.6 Software Configuration Management. The purpose of configuration management is to identify the configuration at a given point in time (referred to as a baseline) and to systematically control changes in order to maintain the system's integrity and traceability (Bersoff et al, 1980:20). This discipline requires that the process be documented and followed religiously (Weber et al, 1991:L2-71 through L2-72).

Once again, the DoD acquisition guidance requires that a configuration management program be established in order to identify the configuration items, control changes to them, record their configuration, and to audit the actual configuration and its identification (5000.2, 1991:9-A-1). The guidance further states that the configuration management program should be consistent with the program's size, criticality, and complexity (5000.2, 1991:9-A-2).

The program office's influence in the Software Configuration Management (SCM) function is most powerful earlier on. Proper CM requirements need to be placed in the contract or SOW (Marciniak and Reifer, 1990:91). It is not sufficient to require the contractor to use "sound configuration management practices;" verifiable requirements need to be written covering the contractor's CM organization, plan, tools, and personnel (Marciniak and Reifer, 1990:91-92). If these requirements are contracted for, the SPO's CM function is simply to verify the contractor's CM efforts and report problems to senior management (Marciniak and Reifer, 1990:92-93).

Similar to SQA, the SPO's SCM role is to contract for and oversee the proper implementation of the contractor's SCM. Again, the question must be asked: are these activities worthy of their own SAMF KPA? No, the same reasoning applied to SQA above applies to SCM. The SPO's CM activities apply to all parts of the system and are not software specific. Therefore, SPO's SQA contracting and monitoring activities can be incorporated into the requirements management and software project tracking and oversight KPAs.

3.7-2 Maturity Level 3. The KPAs for level three are: organization process focus, organization process definition, training program, integrated software management, software product engineering, intergroup coordination, and peer reviews. The following paragraphs address the relevance of these topics to acquisition.

3.7-2.1 Organization Process Focus. The essence of process focus is ensuring that everyone involved in the process understands the process steps, recognizes its strengths and weaknesses, and coordinates to improve the process. A group should be established in order to define and document the standard process for the organization. The members of the group should have a great deal of experience in the field and should understand the organization's process improvement needs. As with all aspects of process improvement, this group needs to be given adequate resources to study and document the organization's process and to study the effects of changes to that process (Weber et al, 1991:L3-1).

This concept is very applicable to the software acquisition process. A Software Acquisition Process Group (SAPG) should be organized at each AFMC product center to improve its acquisition process.

3.7-2.2 Organization Process Definition. The first priority of the group's established organization process focus is to provide a process definition. A standard process is defined, documented, and maintained in order to stabilize the organization's process. Once the process is stabilized, it can be analyzed and improvements can be implemented. Data providing the results of the changes must be collected and analyzed to guide the process improvement. Also, products from previous efforts are archived to provide a library of past successes and failures. This library can be used to guide process improvement as well as provide information for future projects (Weber et al, 1991:L3-11).

Defining the software process is the first step towards bringing the process under statistical process control (Humphrey89:3). Using Lord Kelvin's axiom:

When you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory

kind; it may be the beginning of knowledge, but you have scarcely in your thoughts advanced to the stage of science. (as quoted in Humphrey, 1990:3-4)

The definition provides the means to measure the process and to gain meaningful knowledge of it. It also provides operational definitions of the process which allows all individuals to understand the meaning and purpose of the different organizations and tasks to be performed (Humphrey, 1990:247).

Similar to organization process focus, organization process definition is applicable to the software acquisition process. This KPA implies each product center/SPO implements a standard software acquisition process.

3.7-2.3 Training Program. A training program provides the staff with the knowledge they need to perform their jobs. Furthermore, they may be provided with opportunities to improve their professional education as well. The goal of the training program is to adequately prepare the staff to perform their duties (Weber et al, 1991:L2-23).

In order to run smoothly, the training plan should be documented so everyone in the organization understands their obligations and opportunities. Furthermore, the program should be policed to ensure everyone is provided with the appropriate training (Weber et al, 1991:L2-23 through L2-24).

The need for a well-trained acquisition work force has been recognized by the DoD. The DoD believes that well-trained individuals are needed in order to improve the acquisition process (5000.1, 1991:1-6). This KPA is indeed applicable to the Air Force acquisition process.

3.7-2.4 Integrated Software Management. The process defined in organization process focus and documented in organization process definition provides the basis for integrated software management. All projects undertaken by the organization are managed according to the process definition in order to achieve their objectives. To be useful, the definition must be tailorable, that is be flexible to change so the process can be applied to virtually all situations the organization may encounter. The process definition must also integrate the technical and managerial issues that will be faced on the project. If project objectives are not being achieved, the process definition may be further tailored or even adjusted to correct the problem or oversight. Again, data from all projects is maintained to provide information for future projects (Weber et al, 1991:L3-33).

Establishing and integrating a process is very applicable to the software acquisition process. The SPO's must integrate the standard acquisition process to their program to meet its specific needs. This is a key theme of DoDI 5000.2 (Cochrane, 1991:30).

3.7-2.5 Software Product Engineering. Software Product Engineering covers the activities performed to produce a working software system from the operational specifications or the

contractual requirements. These activities use state-of-the-practice tools and methodologies to analyze the system requirements, produce an architectural design, and implement the design into working code. Also included in these activities are evaluations and inspection processes that ensure the product is produced correctly and that the correct product is produced. The procedures used in the production should be documented in such a way to be useful to every project the organization undertakes (Weber et al, 1991:L3-57 through L3-58).

The government acquisition guidance mandates that software engineering techniques be used in the production of the software system. The guidance also mandates that the SPO ensures that the contractor understands the scope of the effort and is capable of developing the product using state-of-the-practice engineering activities (5000.2, 1991:6-D-1-1).

As in SQA and SCM, the software product engineering activities are performed by the contractor. Once again, it is the SPO's responsibility to ensure proper software product engineering is contracted for and implemented during the program. These contracting/oversight activities could be integrated into the SPO's requirements management and software project tracking and oversight functions.

3.7-2.6 Intergroup Coordination. Just as it sounds, intergroup coordination is the disciplined interaction of all groups working on the project. Every individual in every group understands the system goals and how he or she fits into achieving these goals. Furthermore, every group should understand the project responsibilities of all other groups. The formal interfaces between the groups must be well established and documented. Management provides an atmosphere that promotes a team effort (Weber et al, 1991:L3-79).

Intergroup coordination is indeed a difficult task. Without it, a program may be doomed. With good coordination, the program stands a chance of succeeding (Marciniak and Reifer, 1990:190). However, knowing who is responsible for what and what the official communications channels are is not enough. Other very important aspects of coordination are the informal networking channels, electronic communications (e-mail for example), project telephone listings, meeting reports and minutes, and possibly, a program newsletter (Marciniak and Reifer, 1990:191-192).

The most important aspect of intergroup coordination is management support. The program must operate under an air of trust and communication. Issues should be directly addressed and solved, not avoided or tabled. It is important to remember that software development and acquisition are people intensive activities. Communication and trust are important to the success of the program and should be attained (Marciniak and Reifer, 1990:190). Considering a number of groups (i.e. functional

directorates) within the program office, intergroup coordination is indeed very important and applicable to the acquisition process.

3.7-2.7 Peer Reviews. A peer review is a process that uses developer's peers to examine his products in the hope of detecting problems earlier. Peers are used as reviewers to lessen the anxiety an individual may feel when exposing his work to the public. A methodical, documented process is used in reviewing the products and corrective action plans are followed. The secondary goal of this process is for individuals to become more productive and efficient in their work (Weber et al, 1991:L3-89).

As stated above, peers are employed as reviewers to drive out the developer's fear of failure. In fact, management should not even be present during the reviews (Marciniak and Reifer, 1990:173). In government acquisition, the program office serves in the role of manager. Therefore, it is not appropriate for the government to conduct these reviews.

However, the government could conduct these reviews on its own products, the SOW, for example. It may be useful to employ a review by other acquisition professionals prior to release of a Request for Proposal (RFP) or the signing of a contract or SOW. So, the process could be used by the SPO on its own products (Marciniak and Reifer, 1990:76). Yet, these informal "SPO product" peer reviews would not be very different than the formal government review process already established for most SPO products. Instead, informally coordinating SPO products prior to their formal review should be performed as a standard practice to ensure an effective formal review process. This should be a key practice of many different KPAs such as contract management and software product planning. Therefore, peer reviews should not be established as a SAMF KPA.

3.7-3 Maturity Level 4. The KPAs for level four are: process measurement and analysis and quality management. The following paragraphs address the relevance of these topics to acquisition.

3.7-3.1 Process Measurement and Analysis. When the process is understood (as it would be if an organization achieves process focus, process definition, and integrated software management), it can then be measured and compared against itself. Through these comparisons, more knowledge can be gained as to what the causes of variation are. The organization can then take steps to reduce the sources of variation. The less variation, the more predictable and efficient the process becomes. Therefore, the first step for improving a process is to define it, measure it, analyze it, and reduce the causes of variance. At this point in time, all that remains to be improved is the process itself. The organization can now improve upon the process itself (Humphrey and Sweet, 1987:3-4; Deming, 1982:315).

Process measurement and analysis are the logical extensions of process focus, process definition, and integrated software management. The process has been developed, written, and applied throughout the organization. The organization's performance must now be measured and continually analyzed. Through the study of the process data collected, the relationship between the process and the productivity, quality, and schedule of the process should become understood. The goal of this KPA is to bring the process under statistical process control. The measurements should be analyzed to determine special causes of variation and then the impact of these special causes should be controlled. The process should become controllable and predictable in quantitative terms (Weber et al, 1991:L4-1).

The DoD acquisition guidance states that the contractor should establish "a software process maturity model and process improvement plan" (5000.2, 1991:6-D-1-1). Furthermore, the contractor management should "foster continuous improvements in the software development process" (5000.2, 1991:6-D-1-2). The guidance also requires that the program office collect software metrics and management indicators of the development process (5000.2, 1991:6-D-4). It is indeed clear that the guidance stresses the improvement of the contractor's development processes. It seems only reasonable that the government process used to acquire the product be similarly improved.

3.7-3.2 Quality Management. The reasons behind the government's push to improve the contractor's development process are to eventually improve the product (5000.2, 1991:6-D-1-1). It is management's responsibility to lead the organization toward quality improvement (Humphrey, 1990:335). A lackluster management shows the workers that when trouble comes, management will likely take the quick and easy way out, instead of striving for the quality product. This approach generally does not produce quality software products (Humphrey, 1989:335). The management must commit to producing quality products.

Humphrey suggests the following four, basic quality principles an organization must live by:

1. Unless you establish aggressive, quality goals, nothing will change.
2. If these goals are not numerical, the quality program will remain just talk.
3. Without quality plans, only you are committed to quality.
4. Quality plans are just paper unless you track and review them. (Humphrey, 1990:336)

These four principles are the heart of the quality management KPA. Measurable project goals are established through interaction with the customer and users. Measurable organizational goals are established and progress tracked. Software plans and processes address the organizational and project

quality goals. Finally, the quality process is managed through the use of process measurements and indicators (Weber et al, 1991:L4-15).

In acquisition, simply to require a "quality product" is insufficient. The requirements must be concise, measurable, and verifiable. Planning for a quality program is very important. The organization must determine how it wishes to specify the quality requirements, it must write the requirements in contractual language, and it must be able to measure compliance. These activities should be completed before the RFP is issued. Once the RFP is issued and a vendor selected, the process is still not completed (Marciniak and Reifer, 1990:217-219).

The management of both the seller and the buyer must understand and agree to the quality requirements. Both must unite and present a unified front for enforcing the quality standards and producing a quality product. It is up to the contractor to produce the necessary quality development plan. The program office must concentrate on verifying compliance and providing management support to the contractor's quality function. By working together, the team improves its chance of producing a quality product (Marciniak and Reifer, 1990:228-229).

The federal government does indeed stress the importance of developing quality products. As stated in DoDD 5000.2, "Quality shall be emphasized. It shall be integrated throughout all elements and activities of a program" (5000.2, 1991:6-P-1). The focus of the mandate is on the quality of the design, the quality of the conformance to the specifications, and the fitness for use. Quality products shall be obtained by selecting vendors that produce quality products and by assessing the vendor's management commitment to quality (5000.2, 1991:6-P-1-3).

Once again, the guidance stresses that a contractor provides a management environment conducive for quality development. A similar acquisition environment should help the Air Force manage contractors in such a way to provide an environment conducive to quality production.

3.7-4 Maturity Level 5. The KPAs for level five are: defect prevention, technology innovation, and process change management. The following paragraphs address the relevance of these topics to acquisition.

3.7-4.1 Defect Prevention. At the most mature level, the optimizing level, the emphasis is on constantly improving the process. The goal is to reduce the possibility of an error being introduced anywhere in the process. Defect prevention entails preventing errors, once they are identified and their source known, from entering the system again (Humphrey, 1990:387). In preventing defects from entering the product, the process becomes better and more efficient because less time is spent fixing

the few errors that did find their way into the product (Humphrey, 1990:364). In undertaking a defect prevention program, it is important to remember several points:

1. Feedback is an important part of the Defect Prevention process.
2. There is no single cure-all that will solve all of the problems.
3. Process improvement must be an integral part of the process.
4. Process improvement takes time to learn. (Humphrey, 1990:367-368)

Humphrey also lists the steps involved in defect prevention as being: 1) defect reporting, 2) cause analysis, 3) action plan development, 4) action plan implementation, 5) performance tracking, and 6) starting over (Humphrey, 1989:368-369). Here again, defect prevention takes a commitment from management. Moving toward a defect prevention program is a fundamental change in direction for most organizations. This change does not occur quickly nor does it occur by itself. Management must be the driving force behind the change (Humphrey, 1990:387).

The CMM lists only one goal for defect prevention, that of eliminating sources of product defects once they are identified. This defect prevention policy should be written down in a defect prevention program plan that should be reviewed on a regular basis. The CMM also states that management commitment is essential to the program's success (Weber et al, 1991:L5-1).

DoDD 5000.2 states that the quality emphasis during the engineering and manufacturing development and the production and deployment phases shall be on defect prevention, not defect detection and correction (5000.2, 1991:6-P-3). Again, the government is placing the emphasis on improving the development process. The acquisition process must be able to improve at a comparable pace to be able to adequately manage the program.

3.7-4.2 Technology Innovation. Technology innovation involves analyzing new ideas, tools, and methodologies as they become available. A determination is made as to whether the new ideas could be integrated to improve the organization's process. Pilot programs can be carried out by particular programs or organization sections to aid in this determination. The driving force behind this KPA is to improve the organization's process. New technology should be able to integrate into the process in an orderly and efficient manner. New methods or tools should not detract from the organization's quality; it should enhance it (Weber et al, 1991:L5-17). The Air Force software process needs technology innovations to achieve process improvement.

3.7-4.3 Process Change Management. Finally, continuous process improvement implies continuous changing of the organization's process. The continuous change must be managed. Process improvement goals must be set, action plans must be drawn-up and implemented, and progress

towards the goals must be tracked. The organization's staff and management must be able to cope with this environment of continuous improvement and must be motivated and capable of handling it. In fact, the entire organization must be dedicated to the principles of continuous improvement in the way the organization works. This environment is difficult to foster; management must work diligently to gain everyone's support and cooperation (Weber et al, 1991:L5-31).

DoDD 5000.1 and DoDI 5000.2 require program offices to manage their programs with a disciplined approach. Hence, SPO management of changes to the program's acquisition process is highly emphasized (Cochrane, 1991:34). This implies the process change management KPA is applicable to the acquisition process.

3.8 Other Software Acquisition Key Process Areas

The following sections describe other key process areas that are applicable to a software acquisition process. These topics are in addition to those covered in the previous section about the CMM.

3.8-1 Statement of Work Preparation. After a RFP cycle and a source selection, the system is ready to be put on contract. The contractual vehicle stating exactly what is to be done is the Statement of Work (SOW). This SOW is the basis for communicating management requirements to the developing contractor (Marciniak and Reifer, 1990:74). The requirements contained in this document may cover such things as quality assurance, configuration management, safety, security, metrics, and documentation selection and standards (Marciniak and Reifer, 1990:75). Equally important are the technical requirements of reliability, maintainability, supportability, and test and evaluation standards (Marciniak and Reifer, 1990:75).

Since the SOW is the critical document for conveying requirements to the contractor, it is essential that it be well written. Like the software product, the SOW should have a development process (Marciniak and Reifer, 1990:76). One suggested SOW development process is divided into the following six distinct stages.

Stage one is identifying the people or organizations that need to be involved in the SOW development (Marciniak and Reifer, 1990:76). Many of the buying organization's managerial activities need to be represented along with the users and any independent IV&V or T&E organizations (Marciniak and Reifer, 1990:76). For guidance, some suggested areas of engineering interest are requirements management, selection and definition of CSCIs, tools and environments, and methods (Marciniak and Reifer, 1990:78-88). Suggested managerial topics include documentation, data management, configuration management, quality assurance, and assessments (Marciniak and Reifer, 1990:88-100).

The second stage is the initial preparation for the first SOW team meeting (Marciniak and Reifer, 1990:75). All members of the team should be sent the relevant material needed for developing the SOW. Along with this material, the team members should receive a copy of the SOW development schedule and meeting agenda. Members should come to the initial meeting prepared for the job ahead (Marciniak and Reifer, 1990:76).

The initial meeting is the third stage of the process (Marciniak and Reifer, 1990:75). The meeting should be used to organize the team and to assign each individual/organization appropriate tasking. These tasks should include the conceptual outline of how the software program areas should be contracted in the SOW. Each organization should prepare this conceptual outline for its own area of expertise (Marciniak and Reifer, 1990:76).

The next important stage is to construct a second team, often called a Blue Team, for the purpose of reviewing the SOW (Marciniak and Reifer, 1990:75). The team members should come from both the developing and buying organization's functional areas (i.e. CM, engineering, test, etc.). However, the senior team members should only come from the buying organization's functional areas. The team's goal is to improve the SOW, decrease the SOW preparation time, and help bind the management teams from the buyer and the seller (Marciniak and Reifer, 1990:76).

Stage five is the second meeting between the SOW development team and the Blue Team (Marciniak and Reifer, 1990:76). At this meeting, the conceptual approaches are discussed and finalized. The results of the meeting should be guidelines for the actual writing of the SOW areas. After this time, team members should begin writing their draft inputs for the SOW (Marciniak and Reifer, 1990:76).

The final stage is incorporating the draft inputs from the team members into the actual SOW (Marciniak and Reifer, 1990:76). These activities should adhere to the schedule initially sent out to all team members. The draft SOW is then reviewed by the Blue Team and comments are prepared. The SOW team takes the appropriate corrective actions for the SOW finalization (Marciniak and Reifer, 1990:76).

The procedure described above is an example of a process that can be used to develop a SOW. This should not be viewed as the only way to prepare a SOW. However, many topics brought out by this procedure are valid in any process. These topics are: who develops the SOW, who reviews it, what areas need to be covered in the SOW, how the final draft is produced, etc. What is important is that a standard process is followed.

3.8-2 Risk Management. Early identification of risky software program areas is essential to developing a strategy for resolving the risk and lowering the probability of a "show-stopper" later in the project (Boehm, 1981:13). A secondary advantage of a risk management function is to identify and address new risk items as they arise.

Another reason to implement a risk management program is because DoDI 5000.2 requires it of the SPO and contractor. The program must identify and control risk in the areas of schedule, cost, and performance. The risk management program shall entail the process of planning, identification, assessment, analysis, and reduction of the program risk (5000.2, 1991:5-B-1 through 5-B- 2). The contractor is further required by DoD-STD-2167A to document its risk management process (2167A, 1988:9).

Not only has risk management been a key variable in past efforts to model the software acquisition process (see section 3.4-1), but, as outlined above, it is required by regulation. Risk management is an integral part of the software acquisition process and warrants its own KPA.

3.8-3 Data Management. "Visibility on progress is assessed through different means. Data provides those means" (Marciniak and Reifer, 1990:91). In order for the buyer to have management oversight, there must be some mechanism to provide insight into the program's progress. Engineering data provides the means. Furthermore, government software development standards require the contractor to document the development steps, activities, tools, and information used to create the software products in Software Development Files (SDF) (2167A, 1988:14). Therefore, the contractor is required to record engineering data about the development process.

Still, to be most effective, the contractor should have a process for recording, storing, and controlling this engineering data (Marciniak and Reifer, 1990:91). Like other processes, this should be formalized and documented. The documentation should describe the process and how it's used to support the overall software development process (Marciniak and Reifer, 1990:91).

Some possible evaluation criterion for judging the contractor's capabilities to manage data are:

1. Does the seller understand the importance of engineering data to the development process?
2. Does the seller have a plan to manage engineering data? Does the plan identify all data to be generated on the project and provide for recording, storing, and controlling the data?
3. Does the seller understand the use of software development folders to house engineering data throughout the project?

4. Does the seller provide for the integration of engineering data with the documentation products of the development effort? (Marciniak and Reifer, 1990:91)

These questions could be used to give the government confidence that the contractor may be able to manage engineering data. It is important to note that DoDD 5000.2 requires the program office to ensure that the government has access to all data necessary to support the system throughout its life cycle (5000.2, 1991:9-B-2 through 9-B- 3). Furthermore, past research has indicated timely data is essential for a successful software acquisition program (see section 3.4-1).

The program office itself must also understand the importance of engineering data. This process can indeed lower the cost of the product as well as provide valuable visibility into the product (McKissik and Price, 1979:5). The program office must also understand that inconsistency in implementation of engineering data procedures is a major problem and can negate any positive effects (McKissik and Price, 1979:5). The buyer must be aware of the importance of data management and must be willing to enforce it. Hence, data management is a key software acquisition process area.

3.8-4 Software Supportability Planning. The majority of programming today is supporting software or making changes to keep the current system running (Corbi, 1989:294). If the transition from development to operational support is not well planned, the system may be in trouble (Marciniak and Reifer, 1990:246). The contractor may be required, by DoD standards, to provide transition support by producing maintainable code, transition planning and support, and operational documentation (2167A, 1988:18).

However, before transition support can be contracted, the buyer must decide what the support concept will be (Marciniak and Reifer, 1990:246-7). Some possible concepts are that the development organization provide continuing support, a warranty period may be set up with the developer, or a new supplier be contracted after development, or the user may take over the support function (Marciniak and Reifer, 1990:246-7). With each option comes certain special requirements. For example, documentation requirements could be less stringent for a contractor who will provide for continued support throughout the life of the system than if the users were to take over the support function (Marciniak and Reifer, 1990:248). Whatever the option, the buyer must plan for the inevitable transition (Marciniak and Reifer, 1990:246).

Government guidance requires that the SPO conduct a logistic supportability program to ensure that the product is supportable when fielded. This program shall be disciplined and unified and will apply throughout the entire program. Post-production support planning shall be addressed early in the system life cycle and revisited throughout the program. Specifically, the Computer Resources Life Cycle

Management Plan shall be prepared in close coordination with the Integrated Logistics Support Plan to ensure supportability issues are addressed (5000.2, 1991:7-A-2 through 7-A-3).

3.9 Summary

A process model defines the activities that constitute the process, the activities' interrelationships, and their effects on the system. Once a process has been modeled, the state of the process can be measured disclosing its maturity. Through knowledge of the process' maturity, improvements can be made. The SEI's Capability Maturity Model is an example of such a process model. The CMM is designed around a framework of five maturity levels. Each maturity level contains Key Process Areas that define the salient process activities necessary for successful software production.

The CMM is employed by the Air Force and the software industry to evaluate/assess a firm's software development maturity. Thus, the defense industry's focus to date has been on measuring and improving the software development process. Little attention has been given to the process that drives the requirement for software development, namely the software acquisition process. The Air Force must improve its software acquisition process in concert with its contractor's software development process.

Past research suggests modeling the software acquisition process is feasible. Moreover, the literature indicates the majority of the SEI's CMM framework is applicable to the Air Force software acquisition process. Specifically, 13 of the 18 CMM KPAs are applicable to the acquisition process. The five KPAs not applicable are: subcontract management, software quality assurance, software configuration management, software product engineering, and peer reviews. Four additional KPAs are also required to fully define the framework for the software acquisition process. These 17 KPAs make up the baseline SAME.

IV. Analysis

4.1 Introduction

This chapter draws together the two elements of the research effort, the literature review and Delphi survey. A general framework is presented for the baseline SAMF derived from the literature review. This framework provided a basis for the two rounds of Delphi questionnaires. An in-depth discussion of the two questionnaires is presented along with an analysis of the experts'[†] responses. The experts' answers and comments to the questions provide the underlying structure and content for the SAMF. This chapter provides a complete understanding as to what the proposed SAMF consists of and how it may be employed, according to the literature and the experts' opinions.

4.2 Baseline SAMF—Delphi Questionnaire Development

A thorough literature study relating to the software acquisition and the DoD/Air Force acquisition process provided a great deal of information about important aspects of software acquisition. In addition, information on the applicability of the CMM KPAs was found. By combining the data, the researchers developed a candidate structure for the SAMF. It is this candidate structure that formed the basis of the Delphi questionnaires. The development of the baseline SAMF and how it related to key sections in the Delphi questionnaires is discussed in the following sections. Table 4, page 56 presents the baseline SAMF's structure for reference during the following sections.

4.2-1 Organization Level. The first decision of the research was what Air Force organizational level the SAMF should be applied to. Since the SEI's CMM software process assessments are targeted at the software development organization level, a definition of the target organization level for the SAMF was needed. As the researchers saw it, there were four possible options:

- Target the SAMF at the SPO level where the SPO is the **organization** and the segments/sub-systems/teams are the **projects**.
- Target the SAMF at the SPO level where the SPO is both the **organization** and the **project**.
- Target the SAMF at the AFMC Product Center level where the Product Center is the **organization** and the SPOs are the **projects**.
- Target the SAMF at the Program Executive Officer (PEO) level where each PEO program grouping is the **organization** and the SPOs are the **projects**.

[†] "Experts" are the Air Force personnel who were surveyed. These people met pre-determined acquisition training and experience criteria (see section 2.4-4).

TABLE 4. Baseline SAMF

Maturity Level	Key Process Area	SEI's CMM	Baseline SAMF
Level 1 (Initial)	Risk Management		X
	Contract Management		X
Level 2 (Repeatable)	Requirements Management	X	X
	Software Project Planning	X	X
	Software Project Tracking and Oversight	X	X
	Software Subcontract Management	X	
	Software Quality Assurance	X	
	Software Configuration Management	X	
	Data Management		X
Level 3 (Defined)	Organization Process Focus	X	X
	Organization Process Definition	X	X
	Training Program	X	X
	Integrated Software Management	X	X
	Software Product Engineering	X	
	Intergroup Coordination	X	X
	Peer Reviews	X	
Level 4 (Managed)	Software Supportability Planning		X
	Process Measurement and Analysis	X	X
Level 5 (Optimized)	Quality Management	X	X
	Defect Prevention	X	X
	Technology Innovation	X	X
	Process Change Management	X	X

This question of which organization level the SAMF should target, was a critical decision that would influence the selection of the SAMF KPAs. Therefore, the researchers let the experts decide which level to implement the SAMF. Accordingly, the experts were requested to recommend one of the four options (or one of their own) based on the following information taken from the SEI's CMM: 1) an

organization is defined as "a unit within a company, agency, or service that shares common management, is centered at a single geographical site, and has responsibility for a common business area" (Weber et al, 1991:A-11); 2) the CMM states the scope of the organization may differ depending on the company size and assessment strategy (i.e. evaluating the maturity level of a company's division versus a unit within the division) (Paulk et al, 1991:41); and 3) the methodology for implementing the maturity assessment implies that multiple projects should be evaluated to determine the organization's process maturity (Paulk et al, 1991:38).

For the remaining questions in the questionnaire, one option had to be assumed to assure a consistent reference point for the experts. Therefore, the researchers instructed the experts to assume (when determining the applicability of the CMM Maturity Levels and KPAs) that the SPO was the organization and the segments/sub-systems were the projects (the first bullet in the previous paragraph). The researchers chose this option based on the original idea that the SAMF was to measure the SPO's software acquisition management capability. Therefore, the researchers' logical choice was to target the SPO as the organization.

4.2-2 Applicable CMM Maturity Levels. The next set of decisions was which CMM maturity levels applied to the SAMF. Based on the literature review, the researchers' initial impression was that four of the five maturity levels were applicable to the SAMF. Some evidence led the researchers to believe that Level 2, the repeatable level, may not apply to Air Force acquisition. However, inclusion of the repeatable level depended on what Air Force organization level the SAMF was targeted for. Since the SAMF's target organization level was to be determined by the experts, the researchers let the experts decide which maturity levels would be applicable.

The repeatable level implies that similar projects are undertaken and completed by the same organization. The data gathered in executing these projects allows the organization to further understand the outputs of their process, for example price and schedule. The organization has some level of confidence in its estimations based on this data. If both the organization and the project is defined to be the SPO (the second bullet paragraph presented in section 4.2-1) the repeatable level may not apply. Few SPOs procure multiple systems—each system constituting a "project." The exception is the basket SPO where the acquisition of multiple, similar systems are managed concurrently. The basket SPO's software acquisition process may indeed be repeatable. However, for the SPOs that manage the procurement of single systems (i.e. major programs), the repeatable level does not apply. In this case, a SPO would have to procure multiple systems sequentially over time to be considered repeatable. For example, if the F-15 SPO completed its procurement and remained together to procure the F-22, the

acquisition process may indeed be repeatable. However, most SPOs usually procure only one system (and the sub-systems designed specifically for that system). No evidence was found to indicate that a SPO has ever procured one system and remained intact to procure another. Therefore, if it is assumed that the SPO is both the organization and project, it may be reasonable to omit the repeatable maturity level from the SAMF.

When the SPO is **not** assumed to be both the organization and the project, the argument does not hold. If the definition of the "organization" and the "project" is at different management levels (i.e. the product center is the organization and the SPO is the project), the process may indeed be repeatable. Multiple Operational Flight Programs (OFP) are procured for the same aircraft. Upgrades, Engineering Change Proposals (ECP), and other changes to the original concept could indeed precipitate a repeat of a procurement activity. Therefore, it may be possible to have a repeatable process.

The researchers could not come to a clear decision on this issue and decided to bring it directly to the experts' attention in the questionnaire. A section of the questionnaire requested the experts to choose which of the SEI's CMM maturity levels were applicable to the SAMF. Defense of their opinions was encouraged in the form of comments.

4.2-3 Applicable CMM Key Process Areas. The final set of decisions was which of the KPAs from the CMM should be applied to the SAMF and what, if any, additional KPAs should be added. Unlike the other decisions, a great deal of information was found to help in this decision making process. In the literature review, information relating each KPA to the software acquisition process was found and clear indications of each KPA's applicability was obtained (see section 3.6). The literature also revealed several additional areas that were vital to the software acquisition process (see section 3.7).

In an attempt to be consistent and unbiased, the researchers allowed the experts to choose whether the individual KPAs were applicable. The experts were presented each CMM KPA and asked if it should be included in the SAMF. If the literature indicated that the KPA may not be applicable to the SAMF, the researchers specifically pointed this out in follow-up questions. Furthermore, the experts were asked if the additional areas found in the literature review should also be included in the SAMF. The researchers did not ask the experts to place the additional KPAs in any of the maturity levels. Instead, the researchers used the experts' answers and comments, as well as the information found in the literature review, to guide the placement of any new KPAs.

In the following paragraphs the CMM KPAs that the literature indicated might not apply to the SAMF are described. Included in each is a short discussion of the reasoning and an overview of how questions pertaining to these facts were asked.

Subcontractor Management, a CMM Maturity Level 2 KPA, was the first KPA to be presented with a negative question. The literature review indicated that the Air Force does not have direct control over the subcontractors' actions; they are controlled by the prime contractor. The only exception to this is if the prime contract outlines specific government oversight activities that shall be flowed down to subcontractors. This, however, may not guarantee government control at lower levels. For these reasons, the experts were asked to state their agreement to the following statement: Subcontract Management is a function of the prime contractor. Subcontract Management should not be managed directly by the SPO (see section 3.7-1.4).

The literature indicated that **Software Quality Assurance** and **Software Configuration Management** were product assurance disciplines to be carried out by the developing organization. Furthermore, the government acquisition guidance indicated that the Air Force has an oversight role in these areas during development, not a hands-on implementation role. Specifically, planning for and implementing SQA and SCM is the contractor's job. The SPO's responsibility is to ensure that the contractor properly implements the SQA and SCM plans. It was the researchers' opinion that these oversight functions are key practices of the requirements management, contract management, and software project tracking and oversight KPAs. Hence, the experts were asked to state their level of agreement as to whether these activities were the contractor's responsibilities and whether they should be managed as a separate discipline within the program office (see sections 3.7-1.5 and 3.7-1.6).

Software Product Engineering is a structured engineering process of developing software architectures and designs to meet explicit requirements, and producing the programming product. Seldom does the SPO require a specific software architecture or design to be implemented by a contractor, or for that matter, implement the software design to produce the code. Instead, the SPO oversees the contractor's efforts to ensure he uses sound software engineering techniques to develop and implement the software architecture and design. It is the contractor's responsibility to produce a programming product which meets the contractual requirements. Therefore, the SPO's software product engineering oversight function is a key practice of the software project tracking and oversight, requirements management, and the contract management KPAs. The experts were asked if the software product engineering KPA was a software developer function and not a SPO function (see section 3.7-2.5).

Peer reviews were also found to be an activity of the developing organization. The literature indicated that this process is used to increase the quality of the programmer's code by allowing their

co-workers to review their product prior to its release. Furthermore, the literature stated that the higher levels of management (and in this case, the government) should not be in attendance. The philosophy of peer reviews is applicable to the software acquisition process in the form of government document reviews (i.e. SOW or CRLCMP). However, this philosophy is already integrated into other KPAs such as requirements management and contract management. When developing the baseline SAMF, it was the researchers' opinion that the Peer Review KPA was not directly applicable. The Peer Review question in the questionnaire indicated this position (see section 3.7-2.7).

4.2-4 New Candidate Key Process Areas. In addition to the CMM KPAs, the researchers found several other topics of importance to the software acquisition process. The following paragraphs briefly describe the questions relating to the new areas.

The literature review revealed the SOW preparation had a great deal of affect upon the software acquisition process (see section 3.8-1). Responsibilities the SOW does not specifically task the contractor to do will, in all likelihood, not get done. A great deal of information was found to guide the program office in its preparation. The researchers expanded this area into an even broader topic. Not only is the SOW preparation important, but the process of producing and updating the contract is too. For these reasons, the researchers asked the experts if **Contract Management** was a KPA of the software acquisition process.

Government acquisition guidance was very adamant about **Risk Management**. At every major program review and decision point, the risk management program is assessed. Continuation onto the next program phase is not possible unless the risk management program in place is adequate. The experts were asked if risk management should be included as a KPA (see section 3.8-2).

The government guidance also describes the process for determining the data requirements and how it should be managed. This is so important that **Data Management** is discussed at the Defense Systems Management College (DSMC) program management course. Software is often considered data and, if it is not, its documentation is. Therefore, the process used in identifying and managing data is very important. The experts' opinions on this subject were also solicited (see section 3.8-3).

The final question asked was whether **Software Supportability Planning** should be a KPA of the software acquisition process. The majority of the software life cycle costs is incurred during the operations and support phase. It is also known that decisions made early in the program determine what most of the program's cost will be (as much as 80%). By making the proper decisions early in the life cycle, the supportability costs and, by extension, the system's life cycle costs can be drastically affected (see section 3.8-4).

With this information in hand, the researchers developed the questions for the Delphi questionnaire. The following sections describe the results of selecting the experts and implementing the rounds I and II questionnaires.

4.3 Delphi Survey Implementation

As explained in Chapter II, the researchers employed a modified Delphi technique. Experts in the software acquisition field were sought and their participation was solicited. Two rounds of questionnaires were sent to the experts. Only the responses of the experts who returned both questionnaires were used in developing the SAMF.

4.3-1 Selection of Experts. Selection of the experts was based on the criteria shown in Table 1, page 11. The methods used to find these people were: the ATLAS personnel database, the MCCR Focal Points at the different product centers, information from the SEI, and the SPDP course attendance database. Each method had its unique limitations and advantages. Not every method provided good results or candidates.

While the ATLAS database was useful in locating individuals with a given Air Force Specialty Code (AFSC) at specified bases, information about the individual's experience and CMM knowledge was not available. The SEI provided names of some individuals who had received formal training in the CMM (through the SCE training program) but they could not provide the individual's experience or AFSC. The SPDP attendance database provided names of individuals with CMM experience (the SEI methodology is addressed in several SPDP courses), AFSC, and location. It did not, however, provide the individual's software acquisition knowledge or experience. These methods proved to be of little or no value.

The method that provided the best results and most candidates was the MCCR focal points at the product centers. These focal points provided names of personnel who met all selection criterion. Once initial contact was made, a snowball effect took place. One candidate provided the name of other candidates; the new candidates provided the names of others; and so on. All but a few of the experts were located with this method. The remaining experts were located by means of other personal contacts, similar to the MCCR Focal Points.

In total, 20 individuals were contacted and agreed to participate in the Delphi survey. Of this 20, ten completed both questionnaires. The researchers realize that this may be considered a high mortality rate, but the end numbers were sufficient to allow conclusions to be drawn. A discussion of the reasons for the mortality can be found in section 4.3-4.

4.3-2 Round. Once the experts were identified and agreed to participate, the researchers were ready to publish the first round of the Delphi Questionnaire. The questionnaire was broken into five sections: Demographics, General Applicability, Maturity Levels, CMM Key Process Areas, and New Candidate Key Process Areas. The following paragraphs briefly describe the contents of each section. Appendix A contains a copy of the Delphi questionnaire.

In the **Demographic** section of the survey, the experts were asked to provide their rank, position, AFSC, years of acquisition experience, years of software acquisition experience, SPO functional area expertise, and product center experience. From this the researchers gained insight into where the experts had worked, how long, and in what areas.

The **General Applicability** section asked the experts for their opinion of the SEI's CMM and its usefulness in Air Force acquisition. In addition, the experts were asked if it was feasible to adapt the CMM framework to the Air Force software acquisition process. This information was obtained for two reasons: to validate the premise of the research and to determine if the experts were biased in any way. If the experts did not feel the CMM was a viable tool or if they thought an adaptation would not be useful to software acquisition, the remainder of the questionnaire might not have been favorable.

In the **Maturity Levels** section, the experts were asked if each of the five CMM Maturity Levels applied to the proposed SAMF. As discussed earlier, literary information that could influence the experts' opinion was noted. This section's purpose was to determine the general structure of the SAMF, i.e. its maturity levels.

Next, in the **CMM Key Process Areas** section, the experts were asked to determine if each CMM KPA should be included into the proposed SAMF. Once again, if the literature indicated the KPA might not be applicable, questions were stated to indicate this information. The experts were encouraged to provide comments to defend or support their positions. This type of question was asked to determine the detailed composition of each maturity level. Experts' comments were solicited to provide the information needed to document the goals of each KPA.

Finally, in the **New Candidate Key Process Area** section, the experts were asked if any of the new key topics should be a SAMF KPA. Support was documented to provide the experts the necessary background. Once again the experts' comments were solicited. The researchers did not ask the experts which maturity level each new KPA should be included in. Instead, the researchers used information from literature and government documentation to guide the placement of any new KPA.

The questionnaires were packaged along with a cover letter and a return envelope. Detailed instructions were given and a five working-day suspense was requested. The researchers determined a "drop-dead" date for the return of the Round I questionnaires.

4.3-3 Round II. When the Round I surveys were returned, the researchers prepared Round II. The answers to the questions were compiled and the experts' comments were noted. This information was needed to provide feedback to each expert as to what the others thought.

The Round II questionnaire had only four sections; the demographics section was deleted for obvious reasons. The remaining four sections consisted of the same questions asked in Round I. The experts were provided the group's median (or majority) response for each question. Following this information, the researchers provided all relevant comments (see Figure 2, page 14). For informational purposes, the experts were also provided a summary of their responses to all questions. In summary, the experts were given the question, the group response, and all relevant comments for each question (see section 2.5-1).

The experts were instructed to review the questions and the responses/comments provided. The experts were told to change their response only if their opinion had changed in light of new information. The researchers stated repeatedly that the experts should not change their answers simply because they were not the same as the rest of the group. This second round was done to provide feedback to the experts and to allow them to understand the others' opinions. It is worthy to note that the experts were not told the identity of the other experts.

4.3-4 Delphi Methodology Results. The results of the Delphi methodology were mixed. The 50% response rate as well as problems in the questionnaires' development suggested there were flaws in the researchers' Delphi survey implementation. Yet, the Delphi panel was thoroughly qualified and the survey instrument had high internal validity. The information gained from the experts was of excellent quality and was indispensable for completing this research effort.

Of the 20 experts surveyed only ten completed both questionnaire rounds. The researchers had hoped to achieve higher than a 50% response rate by personally contacting the experts prior to sending out the first round questionnaire (see section 2.4-4). However, the initial contact with the experts was not enough.

One low response rate factor was the expert's availability. Two of the 20 experts identified were within two months of moving to a new job. Due to the difficulty in finding experts, the researchers decided to include these individuals in the sample anyway. Unfortunately, both experts did not respond. Another first round expert was called away to professional military education training in the midst of the survey. Hence, the individual did not get a chance to participate. Expert non-availability was only one factor that contributed to the low response rate, another was the questionnaires' construction.

When developing the questionnaire, the researchers focused on achieving two primary goals: 1) content internal validity, and 2) unbiased questions. Unfortunately, in an effort to achieve these goals, the length of the questionnaire grew too long. To increase the response rate, Martino recommends the number of questions should not exceed 25 (Martino, 1975:59). In stark contrast, the researchers' questionnaires contained 67 questions. Consequently, the length of the questionnaire was most likely a big factor in the low response rate.

Another possible problem caused by the researchers' survey technique was information saturation. Again for completeness and unbiasedness, the researchers supplied the experts all relevant first round comments in the second round survey. However, Dalkey states too much feedback may cause the experts to be saturated with information when answering the questions. Information saturation may degrade the Delphi results (Dalkey et al, 1970:29). The researchers noted that the relative number of comments decreased further into the questionnaire. This was the case for both rounds. Therefore, the researchers felt that the amount of information presented to the experts may have caused the experts to spend less time on the later survey questions. Yet, it is important to note that the caliber of the experts' comments was first rate throughout the questionnaire.

Yet another questionnaire construction problem was the lack of defined terms and simple questions. Some experts indicated they had problems with the semantics of key buzz words such as "monitor," "oversee," and "manage." Others were confused by the way the questions were worded. Some questions were complicated, which confused some experts about the research objective. (E.g. Was the research objective to measure the SPO's software development process or acquisition process?) This confusion may have caused some experts to answer questions based on false assumptions. On the other hand, the nature of the Delphi methodology may have corrected some of these problems. It was evident to the researchers from the first round results that a few experts did not understand the research objectives. Accordingly, the objectives were reiterated in greater detail in the second round questionnaire. Furthermore, the first round results showed the majority of the experts understood the research objectives. Hence, because the experts' quality comments were added to each question, their input served to reaffirm the research objectives. As a result of the additional information provided, the experts' responses to the second round questionnaire reflected their better understanding of the research objectives and each question's meaning.

The selection of the experts can be very difficult even if the criteria is well defined (Brown, 1968:4). In this case, the expert selection criteria was so well defined that it caused the researchers problems in locating individuals who met all the criteria. Nevertheless, a good cross-section of experts from different product centers was selected and the target sample size was achieved. Furthermore, both government

and Air Force active duty personnel were represented on the Delphi panel. Moreover, the experts' backgrounds were excellent. (A complete discussion of the experts' background and experience is in section 4.4-1.) These factors had the net affect of ensuring a high quality Delphi panel which represented the expert population.

As noted above, the survey tool had a few flaws. Regardless, the researchers' goals of content internal validity and unbiased questions were achieved. Questions covering all aspects of the SEI's CMM and the baseline SAMF were asked. The subject matter was thoroughly covered. Therefore, the content internal validity of the questionnaire was excellent. Furthermore, the researchers achieved unbiasedness through a variety of actions. One step taken was to ask at least one open-ended question per candidate SAMF key process area. The open-ended questions gave the experts the opportunity to freely state their opinions. These opinions were fed back to the other experts, thus ensuring that the different points of view were considered by all when answering the final round. The open-ended questions also had the affect of negating any researcher influence caused by the questionnaires construction. Additionally, the questionnaires were pretested for biased or leading questions prior to being sent out. The positive results of these steps were evident in the survey results. The experts reached a consensus on all but one of the questions. (The group's response for one question was split between "agree" and "disagree.")

Overall, the results of the Delphi survey were good. The low response rate as well as the flaws noted in the questionnaires' development were of some concern. However, the researchers felt the quality of the expert panel and the questionnaires' content internal validity overrode these concerns. The data gathered from the survey was central to the development of the SAMF.

4.4 Responses

The experts were very meticulous in answering the questionnaires. All 67 questions were answered and all received comments. The experts took time to understand the objective of the research and to provide valid, relevant opinions and recommendations. The answers to the questions provided the information needed to structure the SAMF while the comments provided the information needed to fill in the SAMF with appropriate "Goals" and "Commitments" to Perform for each KPA. The remainder of this section contains a detailed discussion of the answers and comments provided by the experts. The outline of this section follows that of the Round I survey: Demographics, Applicability of Approach, Framework Structure, Key Process Areas, and New Candidate KPAs. A final section covering general comments and overall recommendations from the experts completes this section.

Please Note: Throughout this section, direct quotes from or summaries of the experts' comments are used. In order to preserve academic freedom and anonymity, the identities of the experts are not used. For this reason, the quotations or references seen in this section do not contain citations.

4.4-1 Demographics. In the demographics section, the experts were asked to provide their name, rank, job title, number of years experience in Air Force acquisition, number of years experience in software acquisition, SPO functional areas they have worked, organizations they have worked for, and any CMM training received.

Space and Missile Center (SMC) proved to be the best source of experts providing five, while Electronic Systems Center (ESC) provided four. The final expert worked for the Air Force Operational Test and Evaluation Center (AFOTEC). Candidate experts were located at Aeronautical Systems Center (ASC) and Human Systems Center (HSC), but none responded to the first round questionnaire. Individuals at the Systems Acquisition School and the Standard System Center were also contacted but failed to reply. However, the experts who did respond had experience in all but one of the AFMC product centers. As depicted in Figure 7, the experts' work history showed that two had worked at ASC. Moreover, three experts had worked acquisition from a MAJCOM and one had worked computer/software acquisition at the Air Force Computer Acquisition Center (AFCAC). Unfortunately, none of the experts had ever worked at ASC-South (Eglin AFB) or at HSC. Therefore, except for HSC, all product centers were represented.

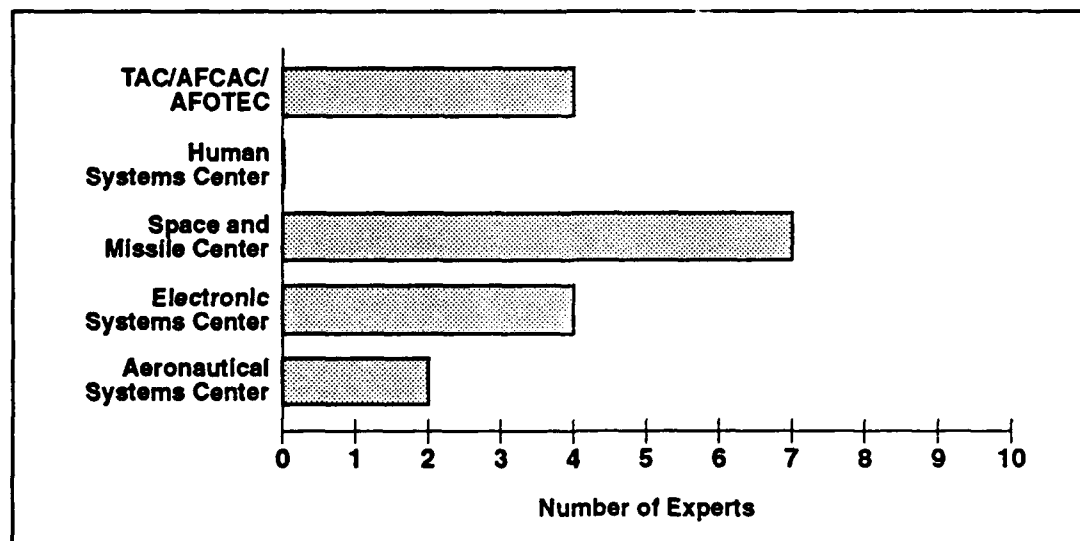


Figure 7. Expert's AFMC Product Center Experience

The experience of the experts ranged from somewhat new to the field of software acquisition to "old hats" at it. The experts' systems acquisition experience ranged from as little as three years to as many as nineteen years with an average of ten years. Their software acquisition experience ranged from three to sixteen years with an average of eight years. The individuals were mostly Captains (four) and GS-13s (three). Also one Major, one GM-14, and one individual worked for a Federal Research Development Center (FRDC), answered both rounds.

In combination, this group has worked in all major program office functional areas. As shown in Figure 8, the majority had experience in program management, software engineering, and software test and evaluation. Half had experience in configuration management and logistics. Three had experience in software quality assurance and contracting. And, several had programming and software maintenance experience. All major functional areas were represented by this group.

Finally, all experts were familiar with the SEI's CMM. As illustrated in Figure 9, page 68, three had actually received SCE training from the SEI. Two individuals had become familiar with the CMM through the SPDP course and three through Computer Resources Acquisition Course (CRAC). More

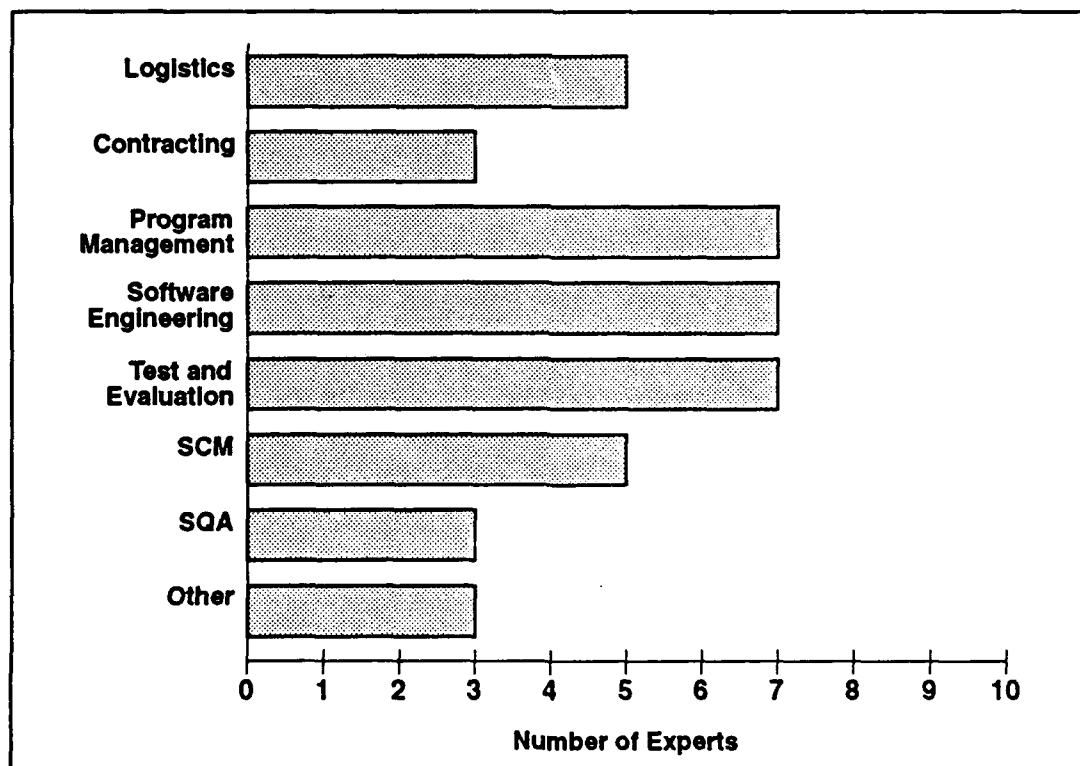


Figure 8. Expert's SPO Functional Discipline Experience

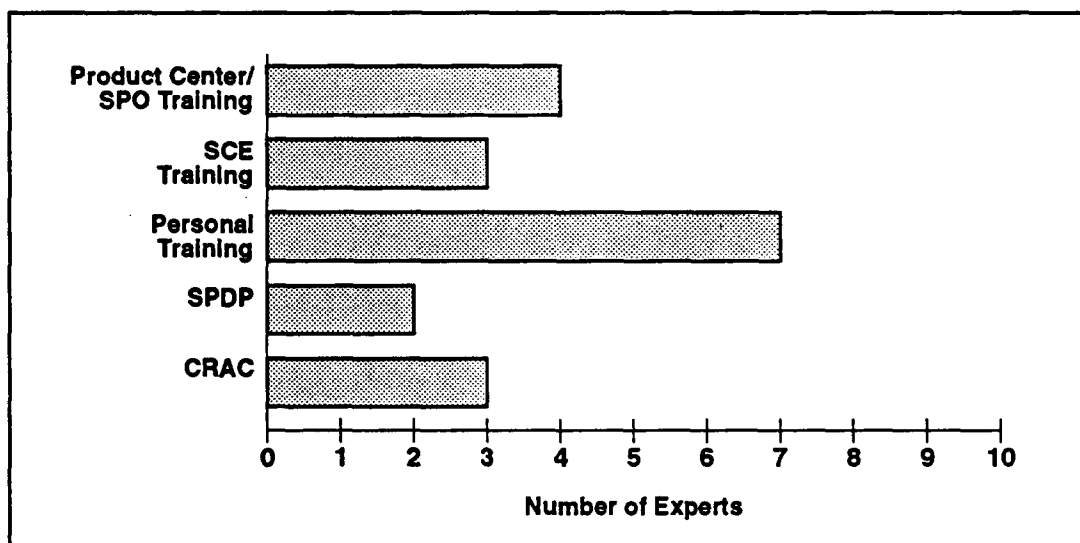


Figure 9. Expert's CMM Training

than half the experts had read Humphrey's *Managing the Software Process*. Also, four had received formal training on the CMM from their product center or program office.

In summary, the experts represented a good cross-section of experience levels and functional areas. All had significant software acquisition experience and understood the CMM. The group did not represent all five product centers. However, the background and experience of these individuals did qualify them to judge the applicability and content of the proposed SAMF.

4.4-2 Applicability of Approach. The first series of questions evaluated the experts' opinion of the CMM and its applicability to the Air Force software acquisition process. Overall, the experts agreed that the CMM was a useful tool to assess and improve the software development process. One expert has "seen dramatic results from our prime contractor's usage." However, others noted problems with its employment. The experts noted that the assessment technique only determines the presence and adherence to documented procedures. The answers to the technical questions of how problems can be solved are not addressed. Furthermore, the results of an SEI's assessments or evaluations are often misunderstood, misinterpreted, or misused. Yet, the group response indicated that the experts believed the CMM to be a viable tool for process improvement.

The majority of the experts believed a SPO could benefit from a similar tool structured for the software acquisition process. "Some tool is essential to measure, report, and make real progress in software management" wrote one expert. Another reported:

Both the CMM and the SAMF are designed to assess the presence of defined procedures. If correctly adapted to the acquisition management domain, the

assessment of the process should be valid. It never hurts to take a look at one's organizational process.

Others indicated that a great deal of valuable information can be gained from such an assessment. This type of approach dovetails with the emphasis on TQM within the Air Force. However, the strongest justification for such an assessment tool came from one expert's negative comment. He noted, "Many SPOs are too lean to make time to do much besides put out fires." This type of activity indicates the management process is very immature and needs process improvement. If this statement represents the majority of the program offices, process improvement could indeed be very useful.

Wary of problems with the SEI approach, the experts stated that there must be confirmed upper level support before a SAMF program could be effective. In addition, the experts recommended all parties involved should be trained in: "1) how to do the evaluation, 2) how to understand the results, and 3) how to improve the process." Also, care must be taken not to emphasize the good scores more than process improvement itself. "The bottom line is that the acquisition itself is more important than...[the] assessment." Furthermore, the tool must be used for the sole purpose of process improvement. In following Dr. Deming's point of driving out fear, the Inspector General (IG) or other such organizations should not be allowed to use the tool for any evaluation or comparison exercise. The experts claimed any use of the tool other than process improvement would undermine the effort.

When asked if the SEI's CMM could be adapted to serve as such a tool, the majority of the experts agreed. One stated, "The CMM is basically a philosophical approach to management. While the questions relate to software development, the fundamental philosophy is still the same." Others stated that the CMM is ideally suited for such a purpose. The industry is continually validating the SEI's assessment methodology through its use. One expert commented, "You can use the basic concept of a capability model but the KPAs would be much different." This comment is a simple restatement of this thesis' research objectives. Again, however, the experts cautioned against several pitfalls in using the SEI's process assessment approach. The experts re-emphasized the need for an in-depth training program and a restriction of the model's use to "an indicator of trends in [process improvement and not]...as an absolute rating scale." Even with these reservations, the majority of the experts agreed with the premise of the research.

In the last part of the Applicability of Approach section were questions covering the Air Force Software Acquisition process and software acquisition within the SPOs. The experts were asked if a standard Air Force process existed, and if so, was it used. Furthermore, the experts were asked if a standard software acquisition process could be applied to all types of SPOs. The final question asked if software acquisition should be a distinct separate functional area within a program office.

In the portion of the literature review focusing on government guidance for software acquisition, the researchers found information which defined portions of the Air Force software acquisition process. In contrast, the group's response indicated that there is no standard software acquisition process defined in the government guidance. The experts stated that the government guidance, "identified only a few important activities," and warned that "every SPO does [software acquisition] differently." The DoD Directives and Air Force regulations/guidance "reflect minimalist controls" over software acquisition. Furthermore, the experts recommended the current software acquisition guidance be changed to reflect problems and issues the SPOs face and not focus as much on contractor issues. However, they also warned against defining a detailed standard software acquisition process at too high a level. They indicated such a detailed definition would be too constraining. An "AFMC standard process will not help the people who work with and within the process, but probably encourages meaningless comparisons between SPOs and product centers." Instead, they stated "the elements of a standard software acquisition process" should be defined. And, software acquisition professionals should be trained in these elements and given the experience (e.g. maintain software at a Air Logistic Center) necessary to manage software acquisition.

The experts' comments listed above complemented the answers given for the question: Can a standard software acquisition process could be employed at all product centers? The experts cited "limited evidence of the use of a standard." They further stated that "differing applications have caused local interpretations to color" the process. A possible explanation was offered by one individual who stated that the product center's engineering support contractors (e.g. Mitre, Aerospace, or TRW) have a great deal of influence in software acquisition. This explanation was supported by another expert who stated that most of these support contractors have little "real-life software development/maintenance [experience]. They know even less than most government organizations think they do." The group's opinion was summarized by one expert who stated:

Within every product center the software acquisition process varies from SPO to SPO. Depending on the experience and training of the acquisition managers, the process implemented will run the gambit from laughable to well-defined and disciplined.

The experts again disagreed on whether or not there was a standard software acquisition process for all program levels. Most comments stated that different types of SPOs (i.e. a major program versus a basket SPO) employed different methodologies when acquiring software. It was said that only the major programs could "sustain the cost of a full military software acquisition." Yet others noted that only smaller programs may indeed be able to employ sound software acquisition practices. In general,

"external influences, environments, and many other (factors) affect what processes exist and which ones are critical."

Furthermore, the group's comments indicated that any process improvement tool must not dictate any single process to the program office. The experts believe the SPO should be given the freedom to tailor the acquisition to fit the program's needs. The consensus opinion was that each program is different and must be treated as such.

In the final question of this section, the experts disagreed that software should be treated as a stand-alone discipline within a SPO. It was noted that "software directly affects logistics, testing, data and configuration management, etc." One individual warned, "Many past software difficulties can be tied to an inadequate understanding of the overall system's requirements and the allocation to software." The experts believed it detrimental to segregate software into its own functional area. Instead, the software should be managed across the functional disciplines. To emphasize this point, one expert gave a poignant example:

When our contract was awarded..., there was one huge software organization. Now, software engineering has been decentralized to work cross-discipline concerns. Communication has really improved and people are working as a team. Non-software people aren't afraid of software anymore!

However, to keep consistency within the SPO, another expert suggested that the SPO appoint a software program manager to coordinate the overall software acquisition process across the SPO's functional lines.

4.4-3 Organization Level. In the next section, the experts were asked at what level the proposed SAMF should be implemented (see section 4.2-1). The majority of the experts indicated the product centers should be the SAMF's target organization, while the SPOs serve as the projects. The vote for the SAMF to target the SPO level was a very close second. The experts in the majority made three key points: 1) The constraining nature of the product center's application domain would facilitate the development of a "reasonably consistent acquisition process;" 2) The SPOs are too diverse and any attempt to define "some standard methodology" at the SPO level would result in too many disconnects; and 3) "At the product center you can learn from all SPOs while leaving them some amount of autonomy to customize as they see fit." On a similar note,

The SAMF should be adapted to product types by taking into account broad product characteristics...Then based upon the model for broad product characteristics, there should be further refinement/tailoring for specific program needs/peculiarities.

Another point was, "To make an impact, the SAMF needs to go beyond the short term objectives of a SPO." This approach was consistent with the Air Force's implementation of the CMM at its Air Logistics Centers (ALC) (Bailor, 1992).

Others did not agree with this point of view. One expert noted the SAMF must be targeted at the program manager or the results would be ignored. Another noted the SAMF should be targeted at Air Combat and Air Mobility Commands and the requirements generation process. Yet another response stated the PEO structure might be a suitable target of a SAMF. These opinions, however, were not in the majority.

In the first section of the questionnaire, the experts validated the use of a tool, such as the SAMF, (provided there is training and restricted use) and stated that an adaptation of the CMM could serve as such a tool. However, this tool should not attempt to impose any single process upon the program offices, but instead it should point out key areas that influence the software acquisition process. The target organization for the tool should be the product centers while the SPOs serve as sample projects. They also called for the SPO personnel to manage software acquisition across function lines and not to segregate software into its own functional directorate.

4.4-4 Framework Structure—Maturity Levels. In the next section of the questionnaire the experts were asked which, if any, of the SEI's CMM maturity levels should be included in the proposed SAMF. As illustrated in Figure 10 on the following page, the majority of the experts stated that all five CMM maturity levels should be included. Furthermore, they stated the general description of the levels, as stated in the CMM, should also be applied. Therefore, the experts indicated the SAMF's general structure should be identical to that of the CMM.

4.4-4.1 Initial. The experts unanimously agreed that a SPO could have an ad hoc process even if it follows the DoD and Air Force guidance. One expert wrote, "There is no way [that] high-level directives can provide sufficient detail for every program." The experts noted that every directive and regulation has an "out" for individuals clever enough to find it. In short, simply adhering to the published guidance (both DoD and Air Force) does not guarantee that a SPO's process has matured beyond an ad hoc level.

4.4-4.2 Repeatable. The experts also indicated that SPOs could indeed have a repeatable process. One expert noted, "Repeatability is essential to post-deployment support, reuse of development and test tools, software maintenance's design recovery, or reverse engineering phase." Another noted that process repeatability has to do with the ability to "formally define and apply a process, not necessarily the same exact one." Others noted that engineering change proposals and Contract Change

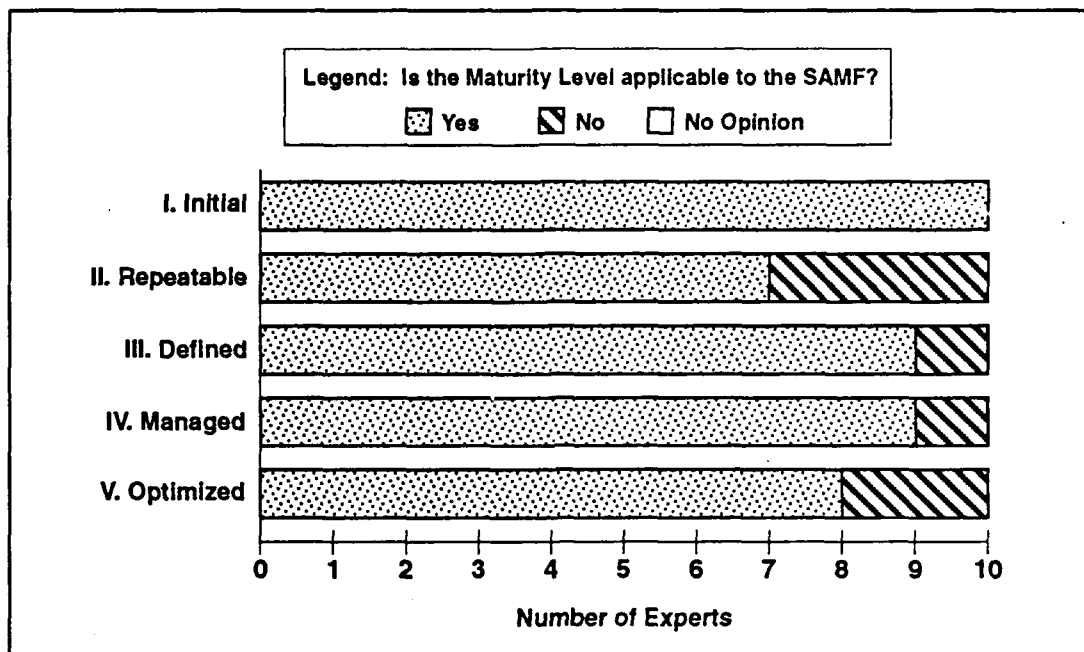


Figure 10. Group Response for CMM Maturity Levels

Proposals (CCP) often take on the air of a “mini-project” which allows the acquisition process to be repeated. Also, if a block-release cycle is defined, each block could be seen as a separate instance of the process (see section 4.2-2). The majority of the experts believed and justified that a software acquisition process could be repeatable, as defined in the CMM.

4.4-4.3 Defined. The experts also concluded the SAMF should have a defined level. However, they reiterated their point that there is no defined standard software acquisition process. One expert stated that under the current documented acquisition procedures (i.e. policies and regulations), many SPOs that have software problems would be “in compliance.” A new set of documented acquisition policies and procedures at product center levels is needed so the SPOs can use and tailor these policies and procedures to define their process. (See experts’ responses to the Applicability of Approach section, paragraph 4.4-2.)

4.4-4.4 Managed. The experts agreed that the managed maturity level applies to the SAMF. However, concern was raised over what the SPO should measure. “Software products” as they relate to managing the software acquisition process, need to be defined. One expert recommended the software acquisition process and product definition, which is measured by the SPO, be a topic for further research.

Nevertheless, the experts agreed this maturity level was important. One expert pointed out that Level 3 only guarantees a "defined" process. An organization at Level 3 could define an incorrect process. However, at Level 4, the organization is attempting to validate its process to determine if the process is indeed appropriate.

4.4-4.5 Optimized. For a SPO to achieve and maintain a Level 5 process, matching commitments from the federal government and the contractor were cited as essential. This is not to say that if a contractor is less than a Level 5 software development organization that the government cannot be a Level 5 software acquisition organization. Instead, the commitments must be in the form of obligations to collect the data. The experts noted that a true optimizing process is costly to achieve, enough to require firm support in the budgeting process. Furthermore, Level 5 processes require a great deal of data collection and analysis. An expert noted that this level of process understanding was not possible unless supported by the contractors.

The experts indicated that process improvements at Levels 3, 4, and 5 also must apply to acquisition as a whole, not simply to the software portion. Others, noting problems that could occur in the inherent data collection tasks at upper levels, asked, "What measures do they use and where do they document them? It's hard enough to find guidance on contractor-oriented metrics." Other experts stated the higher levels may not be attainable by government organizations. Some reasons given were: turnover, training requirements, and budget constraints. Nevertheless, the group response was that all CMM Maturity Levels should remain intact for the SAMF. General descriptions and philosophies of the levels also should remain the same. Therefore, the proposed SAMF would have the same structure as the CMM.

4.4-5 Key Process Areas. Next, the experts were given full descriptions of each CMM KPAs. The experts were asked if the KPAs should be included in the SAMF. If the literature indicated the KPA may not apply to an acquisition process, follow-up questions were worded to indicate this. Comments were solicited from the experts to determine justification and further refinement of the SAMF KPAs.

4.4-5.1 Level 2 (Repeatable) KPAs. As shown in Figure 11, page 75, the majority expert response on all Level 2 KPAs advocated their inclusion in the SAMF. The experts' comments, however, indicated the focus of several KPAs should be changed.

The experts strongly agreed that Requirements Management should be a SAMF KPA. One expert stated, "[It is] the most important KPA in the software acquisition process." The one expert who disagreed with the group's opinion felt the SPO should manage requirements at the system level. "It

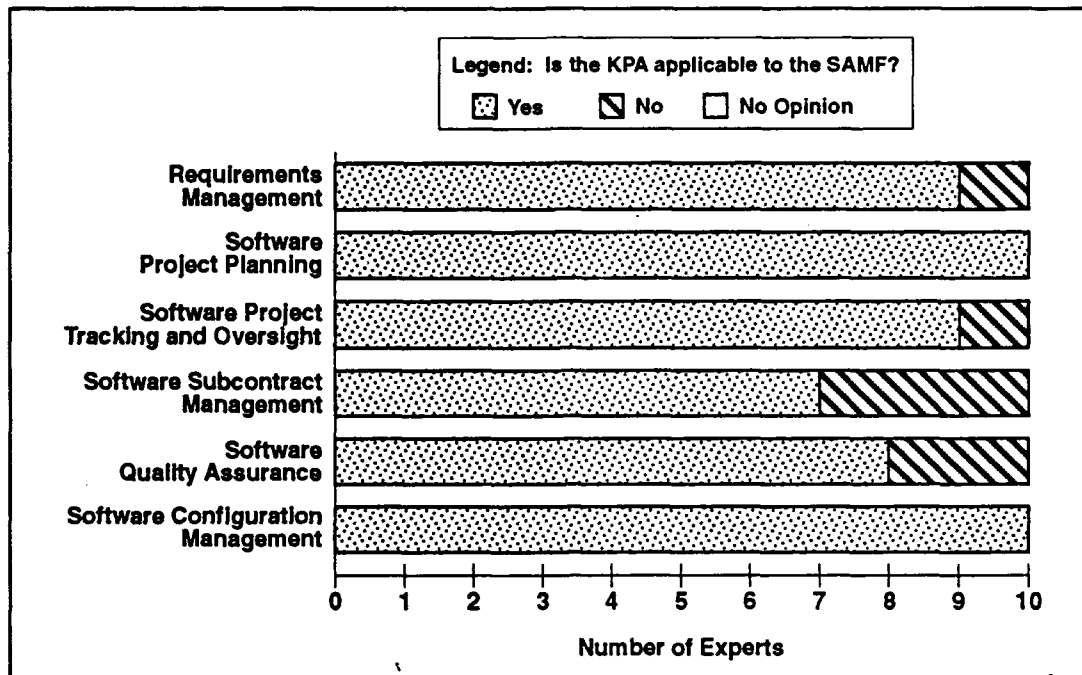


Figure 11. Group Response for CMM Level 2 (Repeatable) KPAs

is the contractor's responsibility to allocate the system requirements to the software." Others agreed with the latter part of this opinion. However, they also felt that it is the SPO's job to ensure the allocated requirements are testable and internally consistent. Furthermore, a defined approach for identification and control of requirement changes is essential at both the system and allocated levels. Therefore, it is the SPO's responsibility to ensure that the contractor's allocation of the system level requirements to software is correct. Furthermore, the SPO must ensure that the maintainers are part of the requirements management process. Lastly, the experts felt that this functional area lacked professional training courses. This KPA was singled out as a very important area of acquisition which needs a formally defined process as well as professional educational opportunities.

The experts noted that the Software Project Planning KPA could "use some tailoring." The program office must "insure that the development process is well defined, reflects good use of state-of-the-art tools and techniques, [and] provides for adequate QA, PA, T&E, etc." Others called for feedback loops, definition of contract measurement criteria, and a Software Risk Management Plan (SRMP). Others noted that software project planning must begin early in the acquisition cycle (i.e. concept exploration/advanced projects stage). The lack of this planning "unfairly limits the [Concept of Operations] and Post Deployment Software Support orientation that is essential" to the program. The software project planning should begin with the program Work Breakdown Structure (WBS),

prior to contracting the WBS/SOW, and should be included in the Systems Engineering Master Plan (SEMP) and Test and Evaluation Master Plan (TEMP). Again, the experts felt that this KPA was very important to software acquisition but needed minor changes to its scope.

The expert's opinion was the **Software Project Tracking and Oversight** KPA should be part of the SAMF. One noted the axiom "that which is not tracked will not be completed." The experts agreed this KPA included tracking and assessing the contractor's accomplishments according to the contract. However, the contract only states the *what* (i.e. what data items are to be deliverables) not the *how* (i.e. how the software development process is to be executed). The SPO's efforts should be focused more toward tracking and assessing the *how* versus the *what*. This means the focus is on the process not the product. Therefore, the contractor's and SPO's efforts should be measured against their plans (e.g. the SDP, SRMP, and Software Program Plan). Yet, the scope of the tracking should not be limited to the documented plans. The objective should be to resolve any process problem, not just those problems that correspond to defined processes. Nonetheless, the experts indicated software project tracking and oversight was an important aspect of software acquisition.

The majority of the experts agreed that **Subcontract Management** should be a SAMF KPA. Those who disagreed pointed out that it is the prime contractor's responsibility to manage the subcontractor, and the SPO should not "get into their knickers!" However, most of the experts agreed with the mind-set of one expert who stated, "The management of important and sometimes critical subcontract elements of the system being acquired must be considered as a KPA." The thrust of the majority opinion was that the SPO needs to achieve visibility into the subcontractor's efforts. This should be accomplished by monitoring them through the prime contractor. To this end, several different opinions were offered to modify the subcontract management KPA. One group suggested the KPA be renamed Software Contract Management. They emphasized the need for the SPO to levy the proper requirements on the prime to allow the government to monitor the subcontractor's efforts. Others suggested it be renamed Software Subcontract Monitoring. These experts noted that it was the underlying purpose of the program office to manage the prime contractor. Therefore, a Software Subcontract Management KPA would be meaningless.

The experts also felt the **Software Quality Assurance** KPA should be included in the SAMF. However, the expert's answers to the follow-up questions appeared to conflict with this opinion. The experts agreed: 1) SQA is a contractor function and should not be managed by the SPO as a separate discipline; and 2) the SPO's software program management should ensure the contractor's SQA function is performed properly. The implication was that the SPO should not perform SQA but SQA should be a SAMF KPA. As it turns out, the experts stated SQA should be a SAMF KPA but its

definition should be tailored to a monitoring function instead of a hands-on review and audit function. The experts also indicated the SQA KPA could be integrated into the SPO's program management or contract management function. They also pointed out that SQA should be kept separate from the software product engineering group. This independence must be maintained to ensure the quality assurance is not adversely influenced by the engineering/development group.

The experts stated the **Software Configuration Management** should be a SAMF KPA. They disagreed with the statement: The SPO should not manage SCM as a separate discipline. However, the experts did agree that SCM is primarily a SPO's Configuration Management (CM) directorate's responsibility. The experts keyed in on the configuration identification and change control CM activities. One expert stated that the SPO's software acquisition management should actively participate in "reviewing and approving changes to configuration items." However, they also stated the government's focus should be on the product baseline, not the developmental baseline. This should include planning for the product baseline's transition to the government. Another expert gave a completely different opinion from all others:

This activity determines the basic components of the product and sets the post deployment software support costs and schedules for the product life cycle. The configuration management elements of this KPA should be performed by the user versus the SPO or contractor.

Nonetheless, the group's opinion was the SAMF should include SCM as a KPA.

4.4-5.2 Level 3 (Defined) KPAs. The experts also indicated that six out of seven Level 3 KPAs should be included in the SAMF, as illustrated in Figure 12 on the following page. As with the Level 2 KPAs, some needed modification.

One expert made a special point to indicate that this level and all above it should de-emphasize software and focus on system. According to this expert, all the higher level KPAs directly relate to the entire acquisition process. Furthermore, he stated that it may be wise to adapt the CMM to an Acquisition Maturity Model as well.

The experts agreed there must be some type of **Organizational Process Focus**. When asked at what organizational level a Software Acquisition Process Group (SAPG) would be most effective, the majority stated it should be implemented at the SPO level. One expert added, "We have such groups [in the SPO] and they actually work!" However, several suggested that SAPGs should be implemented at all levels, including the product center, AFMC, and DoD levels. Another suggested the SAPG should "not be limited to software" and should cover all aspects of systems acquisition and be staffed by the program managers.

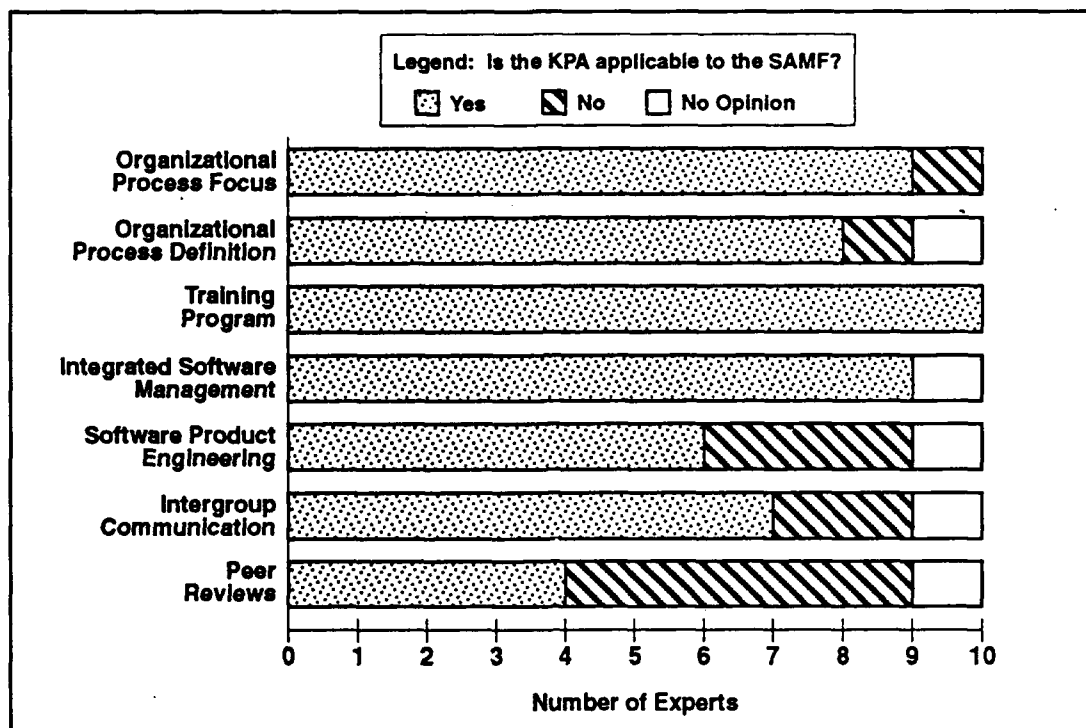


Figure 12. Group Response for CMM Level 3 (Defined) KPAs

The experts felt there should be a **Organizational Process Definition** KPA. Their comments emphasized the same theme in their comments stated in the "Applicability of Approach" and "Organization Level" questionnaire sections (see sections 4.4-2 and 4.4-3). That is, standard acquisition guidelines should be defined at the product center level for **each application domain** (i.e. aircraft, C³I, missiles, etc.). These guidelines should define the acquisition process's elements and include the software specific processes. Then, each SPO should tailor the acquisition guidelines to meet its program needs. As the group's opinion, the SPOs must be allowed the flexibility to tailor the process to their specific program needs. However, all SPOs within an application domain should base their software acquisition process definition on a standard set of guidelines.

The experts unanimously agreed that a **Training Program** was an important SAMF KPA. According to one expert, training was "key to the Air Force's problem." Several experts cited that the Acquisition Professional Development Program (APDP) training was a good starting point, but insufficient to properly prepare the software acquisition professional. Software acquisition managers need training "on the fundamentals of software acquisition and software development." Furthermore, the experts recommended training in the various functional areas of the program office such as

configuration management, test, quality assurance, and others. They also stated the SPO personnel need to be trained on the user's organization, mission, and requirements. One expert suggested, "A lot of the 'users' orientation could be covered by moving/rotating software managers from the MAJCOMs through the acquisition centers." In short, the experts stated the software acquisition professional must be an expert in software development methodologies, and an expert in all aspects of the acquisition process and in the user's environment.

The experts also stated the SAMF should have a **Integrated Software Management KPA**. The experts indicated this KPA applies across all functional areas within a SPO. The SPO, of course, cannot directly affect the contractor's management coordination. However, it can do a great deal to improve the managerial relationship between the contractor and the government. Furthermore, the program manager should ensure a cross-flow of information within his organization.

The next Level 3 KPA was **Software Product Engineering**. Though the SPO does not develop software, the experts recommended the KPA remain in the SAMF. However, one recommendation was to change the CMM KPA's wording from "build and maintain" to "acquire and support." Also, the developmental and engineering tasks outlined in the CMM KPA should be converted to monitoring functions. The experts keyed in on the fact that the "SPO should be actively involved in measuring and evaluating the developer's product engineering process."

Similar to the integrated software management KPA, the experts recommended the program manager be responsible for **Intergroup Coordination**. One expert noted that this aspect "is often missing." Furthermore, one expert pointed out that the DPRO representatives, the customer, and the test organizations must also be remembered.

Finally, **Peer Reviews** was the only CMM KPA that the experts stated was not directly applicable to the SAMF. The experts emphatically pointed out that the government should not take part in the contractor's software developmental-type peer reviews. However, the experts did state the SPO should "mandate [the contractor's use of] peer reviews by contract." One expert noted that ensuring the contractor uses peer reviews also could be accomplished by including this activity in the SDP. The experts further commented that the government's current role in peer review type activities entails very high level activities such as design reviews and audits. Alternatively, some experts indicated peer reviews may be an applicable SAMF KPA if it was defined as a review of program office products such as contracts, SOWs, and correspondence. Yet, others indicated there would be problems implementing peer reviews in the SPO. One expert stated the SPOs would need more personnel to perform the peer reviews. Another expert questioned the integrity of a SPO peer review stating, "Who in the Air Force

(with its accompanying rank structure) thinks a peer review will be unbiased and honest?" The final expert vote was that peer reviews should not be a SAMF KPA.

4.4-5.3 Level 4 (Managed) KPAs. As shown in Figure 13, the experts agreed that both of the CMM Level 4 KPAs should be included into the SAMF. Like the Level 3 KPAs, one expert

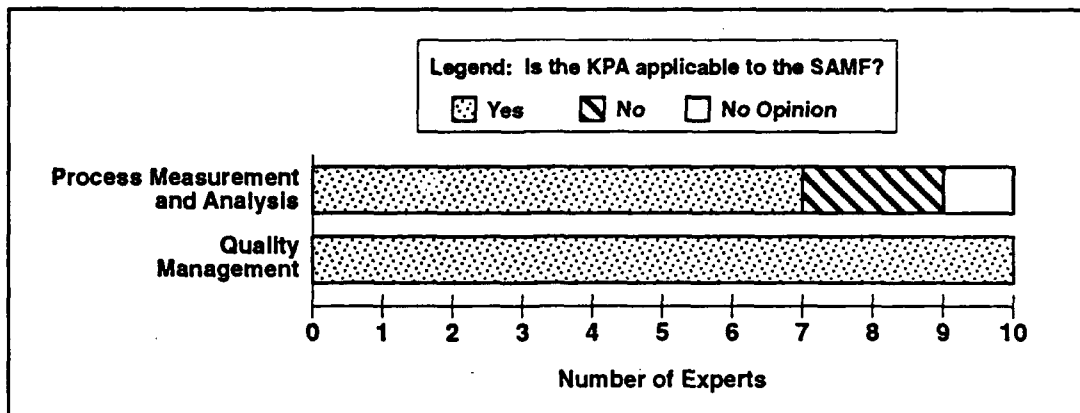


Figure 13. Group Response for CMM Level 4 (Managed) KPAs

continued to note that the higher maturity levels should be applied to the general acquisition process, not simply the software portion of it. However, unlike KPAs from the other levels, the experts did not present any major modifications. The experts did note that in order to facilitate **Process Measurement and Analysis**:

Acquisition process measurement techniques and tools must be developed and maintained at a level higher than the SPO. This would enable the metrics to have value across projects and help SPOs decide which process model to use.

The experts noted that **Quality Management** of the contractor's efforts was likely to occur at reviews and audits than through constant vigilance and effort. They agreed that the philosophy of this CMM KPA holds true for the software acquisition process. However, some experts were confused as to the differences of this KPA and the SQA KPA. They stated that the quality management KPA was really the planning process for SQA. Hence, why segregate this process from SQA? Yet, others indicated the objective of quality management is much broader than just SQA. Quality management is applicable to many elements of the acquisition process.

4.4-5.4 Level 5 (Optimized) KPAs. As in the Level 4 KPAs, the experts agreed all should be included with very few modifications. Figure 14, page 81 depicts the experts' vote. Most of the comments for this KPA level were of a clarification nature, restating the focus in program office terms.

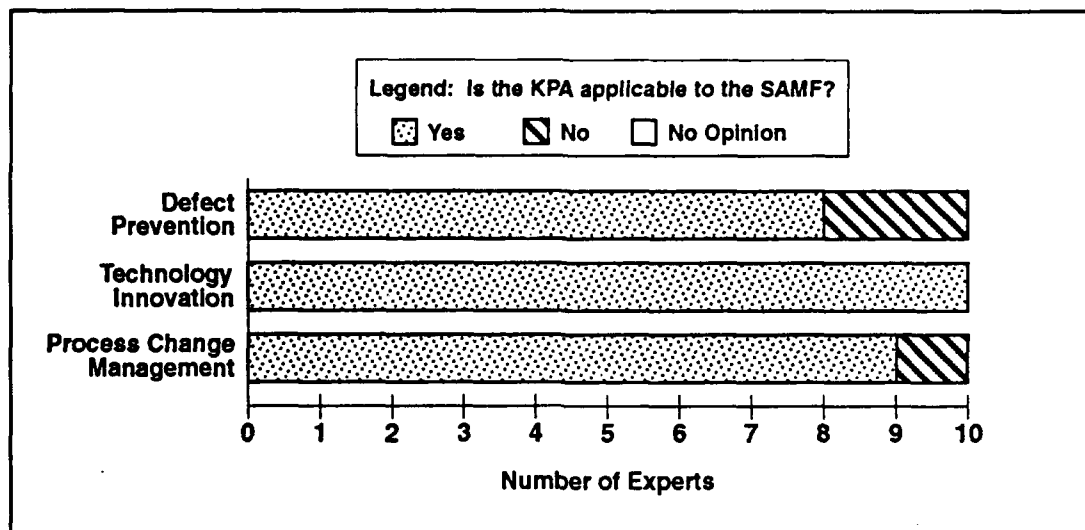


Figure 14. Group Response for CMM Level 4 (Optimized) KPAs

The majority of the experts agreed Defect Prevention should be included. One expert suggested the SPO should practice software acquisition process defect prevention by focusing on "lessons learned" versus tracking the number of requirements and coding defects found.

Software acquisition Technology Innovation was indicated as important to the SAMF. The experts stated that the ideas for the innovation could come from virtually anywhere, ranging from the user, to the labs, to the program office staff. One expert noted that as well as what is stated in the original KPA, the program office also must make the contractor aware of applicable innovations if a situation should warrant it. One expert offered the following summary:

The KPA should be coordinated across AFMC with candidate technologies identified by the interaction between the laboratories, centers, SPOs, and form a basis for an element of continuous process improvement. This should be monitored and reported on at all levels of the Air Force as a TQM initiative between the users and AFMC.

The experts' opinions were to include the KPA with the modifications described above.

The final CMM KPA, Process Change Management, was also included by the experts. One expert stated the objective in the following manner, "It is important that the cost of making change does not outweigh the gains. You must have an efficient means of defining and re-defining the process." Another expert suggested this KPA should be combined with the Organizational Process Focus KPA. However, one expert claimed this effort to be "bureaucratic waste."

The experts indicated that all of the CMM's original KPAs, except for peer reviews, should be included in the SAMF. Several KPAs were in need of modification or re-focusing. Others could be combined. However, all general ideas were deemed to be important to the software acquisition process.

4.4-6 New Candidate KPAs. The experts were then asked to review other candidate key process areas, as indicated by the literature review. As opposed to the CMM KPAs, the experts agreed that all should be included in the SAMF.

A majority of the experts strongly agreed that Risk Management should be a KPA in the SAMF. One stated that this area alone will tend to drive most of the others. Another stated that attention must be paid to Government Furnished Equipment (GFE). He stated that GFE could cause severe problems if not completely understood. For example, the government could be liable for damages to the contractor resulting from problems caused by reusing GFE software components. The program office should take care when planning around/for GFE. Other risk areas noted were schedule constraints (delivery/repair) and operational capabilities.

The Contract Management key topic did receive more favorable votes than unfavorable. However, some experts indicated that this was a duplication of effort, in that the tasks associated with this KPA are already accomplished in others (i.e. software project planning and subcontract management). Another expert stated the SPO puts too much emphasis on making contract modifications "timely and efficient." This causes the SPO to focus more on the contract modification itself and less on the original objectives—namely to incorporate the correct contract change. "This is especially risky in the software arena." However, one expert noted it was important the SPO ensures "that the software tasks have been properly defined, compliance documents have been properly tailored and all necessary deliverables have been requested."

While a majority of the experts strongly agreed Data Management was applicable to the SAMF, there was a strong vocal opposition. They stated this activity had no added value and was nothing more than a bookkeeping or tracking function that added little to the quality of the product. Yet another expert, who strongly agreed with the KPA, made the point, "What if a contractor folds? Without data, [the government] would have a bunch of unmanageable software." As an alternative, one suggested this be combined with the contract management KPA (or where ever that key topic was placed).

The last question in the survey asked if Software Supportability Planning was applicable to the SAMF. Here, as in a few of the others, a majority of the experts strongly agreed that the key topic should be included as a KPA in the SAMF. "Software support planning must be a prime driver for the entire software team...." Specific activities pointed out included the formation of a Computer Resources

Working Group (CRWG) and the publishing and maintaining of a CRLCMP. Open communications with the user were cited as essential for success. It was also noted that more emphasis might be placed in this arena with the advent of the Integrated Weapon Systems Management (IWSM) concept.

In completing the questionnaire, the experts validated the structure and general content of the proposed SAMF. They indicated it should have the same five maturity levels the CMM has as well as all but one of its KPAs. Whereas the maturity levels' conceptual idea remains intact, several of the KPAs were modified or re-focused. In addition, four supplemental KPAs were approved by the experts.

4.4-7 General Recommendations. In addition to the comments provided to the questions, the experts provided several other recommendations and noted other areas of concern. The two main areas were the scope of the proposed framework and its intended use.

As stated in the previous sections, several comments were made that indicated a broader scope may be appropriate. There seemed to be a need for and an interest in a general acquisition maturity framework, one that measured the SPO's process maturity, not specifically one aspect of it (such as software acquisition). Several such comments were discussed in the previous section. As a note to the reader, a further discussion of this topic may be found under the "Recommendations for Further Research" in section 5.4, page 94.

The second area of concern was the intended use of the tool. Initially, the researchers were very concerned this tool may be used for SPO evaluation/comparison. At least one expert shared this concern and wrote:

I strongly feel this is a very good approach (and perhaps essential). With the acceptance of TQM by the Air Force and DoD, this fits in real well. The concern is that it may be used by external staffs (e.g. MAJCOM staff assistance visits, IGs, etc) either directly (where they use the questionnaire) or indirectly (where they use the SPO's answers) as basis for their evaluations. This may cast a stigma or create a reluctance to use this tool.

The authors wish to note that this type of activity violates the fundamental concepts of TQM and the SEI's CMM philosophies. The potential for misuse is real. Care must be taken to avoid any such corruption of a potentially useful tool. A further discussion of potential uses and controls is given in the following section.

4.5 The Software Acquisition Maturity Framework

Both the information presented in Chapter III and the experts' recommendations presented in section 4.4 were used to modify the baseline SAMF. In addition, the goals and the commitments to perform listed in the SEI's CMM were added to the SAMF and modified as indicated by the experts

and the literature. When the two disagreed, the opinion of the experts was used. If neither the experts nor the literature indicated a goal or commitment needed modification, the experts simply modified its statement to reflect a government acquisition management focus.

TABLE 5. Software Acquisition Maturity Framework

Maturity Level	Key Process Area	SEI's CMM	Baseline SAMF	Final SAMF
Level 1 (Initial)	Risk Management		X	X
	Contract Management		X	X
Level 2 (Repeatable)	Requirements Management	X	X	X
	Software Project Planning	X	X	X
	Software Project Tracking and Oversight	X	X	X
	Software Subcontract Management	X		X
	Software Quality Assurance	X		X
	Software Configuration Management	X		X
	Data Management		X	X
Level 3 (Defined)	Organization Process Focus	X	X	X
	Organization Process Definition	X	X	X
	Training Program	X	X	X
	Integrated Software Management	X	X	X
	Software Product Engineering	X		X
	Intergroup Coordination	X	X	X
	Peer Reviews	X		
	Software Supportability Planning		X	X
Level 4 (Managed)	Process Measurement and Analysis	X	X	X
	Quality Management	X	X	X
Level 5 (Optimized)	Defect Prevention	X	X	X
	Technology Innovation	X	X	X
	Process Change Management	X	X	X

4.5-1 Modifications. The following sections describe the major modifications made to the baseline SAMF. Any modification that simply reworded a CMM statement to reflect an acquisition process perspective is not discussed in detail. The final version of the Software Acquisition Maturity Framework is in Appendix C. Table 5, below presents the SAMF structure for use as a reference during the following discussion.

As stated earlier, the experts agreed the SAMF should target an application domain within the product center as the "organization" and the SPOs as the "projects." Furthermore, the experts agreed the CMM's five maturity levels should be mapped directly into the SAMF. Therefore, an in-depth discussion of the levels is not presented.

Per the experts' recommendations, all new candidate KPAs were added to the SAMF. The four KPAs are: risk management, contract management, data management, and software supportability planning. The researchers used data from the literature review to guide the placement of these KPAs into SAMF maturity levels. The risk management and contract management KPAs make up Level 1. Data management was placed in Level 2 and software supportability planning was placed in Level 3. The reasoning behind the new KPA's maturity level placement is explained in the applicable sections below. What follows is a description of the changes made to the KPAs, including the goals and commitment to perform, in each SAMF maturity level.

4.5-1.1 Level 1 KPAs. Early in the questionnaire, the experts indicated that the program office could comply with the Air Force governing guidance and regulations and still have an ad hoc process. The guidance indicated that risk management was required to advance into later stages of the system life cycle. Also, the experts felt that Risk Management was important to understand and control early because it drives several other activities. For these reasons, the researchers determined risk management to be a KPA needed to qualify as a Level 1 organization.

For similar reasons, Contract Management was also placed in the new first maturity level. Contract management is required in order to keep pace with the changing needs and environment found in government acquisition. Furthermore, managing contract changes only allows an organization to understand where it has progressed. Simply managing the changes to the contract does not establish repeatability of the process.

By requiring KPAs to qualify for Level 1 maturity, the researchers have slightly departed from the CMM structure. It is worthy to note that one of the criticisms of the CMM was that there were no qualifications to be an ad hoc organization. In the SAMF, an ad hoc organization understands the risk inherent in the program and manages the major changes to the requirements reflected by contract

changes. The experts noted that this alone does not guarantee a process that is more than ad hoc. Therefore, the researchers developed KPAs needed to qualify for Level 1. This implies the organization could theoretically have a Level 0 process, indicating the organization is not following the minimum DoD policies and regulations.

The goals of a risk management program were to document and continually update a risk management plan. The plan should address the actions necessary to control potentially risky items and to lower the possibility of significant schedule, cost, and operational problems.

Contract management identifies the user's requirement changes and ensures they are reflected in appropriate changes to the contractual documents. A documented process should be followed that ensures the process is as efficient as possible. The process should allow the user to easily identify the requirements for change and for the program office to easily make the appropriate contractual modifications.

4.5-1.2 Level 2 KPAs. The first of the Level 2 KPAs is **Requirements Management**. Both the experts and the literature emphasized the importance of managing the changing requirements of Air Force software acquisition. The experts also noted the need for requirements management training for the software acquisition personnel. Furthermore, the literature noted that both the government (program office and user) and the contractor must agree to the requirements management process and be active participants. The CMM already addressed the need for change management, however, a goal was added to establish formal training. Another new goal added to the documented requirements management process was a contractor/government agreement.

For **Software Project Planning**, all three CMM goals and the two commitments needed slight wording modification to stress acquisition, not development. In addition, the experts noted that review of the contractor's Software Development Plan was essential to ensure its adequacy. The literature noted that the government management plan for the software acquisition effort must be documented in the CRLCMP. Furthermore, the acquisition strategy must reflect this documented management approach. A goal and a commitment to perform to review the contractor's plan was added, as was a goal to produce a CRLCMP.

DoD guidance stated that the program office must perform **Software Project Tracking and Oversight**. Furthermore, the experts stated the SPO must track and resolve an software acquisition/development process problems encountered. However, the goals, as stated in the CMM, do not reflect acquisition priorities. Two of the original goals were modified slightly. However, one goal, which stated management must ensure all personnel understood the software development commitments made,

was deleted. This goal was replaced with one stating the program office monitors the development progress of the contractor. Another associated commitment was added to the goal to indicate that the program office has a written policy regarding the contract and use of management indicators and metrics.

The experts indicated that **Software Subcontract Management** must be modified. The new goals are to gain access to and monitor the subcontractor's status in his software development efforts. The only original goal to remain unchanged was the program office's comprehension of the commitments made between the prime and subcontractors. No new commitments were added.

For both the SQA and SCM KPAs, the wording was changed to indicate a monitoring function. The changes made were suggested by both the experts and the literature. The new SCM goal plans for the transition of the product baseline to government configuration control.

The literature indicated that **Data Management** is a discipline similar to the product assurance disciplines of Configuration Management and Quality Assurance. Therefore, the researchers placed the new data management KPA in Level 2. The goals of data management are to determine all data items needed for the product's entire life cycle, contract for them, and ensure that they are delivered. The identification process should be redone on a periodic basis to ensure the data requirements are still valid.

4.5-1.3 Level 3 KPAs. Although there was concern that this level and all above it should reference the entire acquisition process, the researchers confined themselves to the scope originally stated in Chapter I. Therefore, the following modifications describe only the software aspects of the acquisition process.

The goals and commitments stated for **Organization Process Focus** in the CMM were indicated to be appropriate to the SAMF by both the literature and the questionnaire responses. The only major modification was the establishment of a Software Acquisition Process Group to be convened at the product center level. The experts' response indicated SAPGs should be convened at all organization levels. However, the researcher's choose to target the "organization" level (i.e. product center) to be consistent with the CMM's software engineering process group definition.

The **Organization Process Definition** was adequate with only a slight modification. The literature did not indicate the level at which the process should be defined. However, the experts stated the process should be defined at the product center level for each application domain. The definition of "organization" in the SAMF reflects this recommendation.

The literature indicated that a **Training Program** was very important to the acquisition process. The opinion of the experts was similar. Therefore, the KPA's definition and goals were modified to reflect the acquisition process. However, the experts provided a great deal of information for defining the scope of the training program. This information was incorporated into the commitments to perform.

The experts recommended **Integrated Software Management** remain in the SAMF. However, the only modification suggested was to apply this KPA to the entire acquisition process. The literature revealed no change to what was written in the CMM. Since the research objectives focused on software acquisition, the KPA stood as written in the CMM.

Although the literature indicated **Software Product Engineering** should be deleted from the SAMF, the experts voted to keep it. The wording was changed to reflect a monitoring function and an "acquire and support" activity.

Both the experts and the literature indicated **Intergroup Coordination** was important to the acquisition process. However, neither offered any changes to the KPA as defined in the CMM. Therefore, it was modified to reflect the software acquisition slant.

The **Peer Reviews** KPA was not incorporated into the SAMF. The experts indicated this KPA was essential to the development process, however, the government should not take part in the contractor's peer reviews. The literature also indicated that the government should refrain from participating in developmental peer reviews. The concept of the government performing peer reviews on its product (i.e. SOW, RFP, correspondence, etc.) was not deemed a significant activity to warrant a separate KPA. Therefore, the SPO's peer review type activities were incorporated in other SAMF KPAs.

The final new KPA, **Software Supportability Planning**, was placed in Level 3. Supportability planning implies that life cycle considerations have been taken into account. To completely identify the supportability issues, a defined acquisition process should have been achieved. Without a defined process, any success or failure in the life cycle support phase could be attributed to any one of several factors. When a standard defined process is followed on all projects, the true supportability issues can be identified and controlled. One goal of this KPA establishes a software working group (such as the CRWG) chartered to make recommendations to the program manager on issues regarding software supportability. A second goal is to ensure open and frequent dialogue between the program office and the user's community in order to completely understand the life cycle requirements.

4.5-1.4 Level 4 KPAs. The two KPAs for Level 4 imply that the software acquisition process is measured. To do so, acquisition metrics must be developed and the data maintained.

Therefore, the formation of an acquisition metric program is required as a commitment of **Process Measurement and Analysis**. Otherwise, the remainder of this KPA and all of **Quality Management** remain as written in the CMM. The literature and opinions of the experts support this action.

4.5-1.5 Level 5 KPAs. The KPAs of **Defect Prevention** and **Process Change Management** were accepted as is by the experts. The literature offered no resistance to this, either. **Technology Innovation**, however, was changed slightly. Although the literature offered no modifications, the experts suggested a few. It was noted that acquisition process data should be collected and analyzed at all Air Force levels. Technological Innovations could come from any source, but some organization must be able to determine what is applicable to a given program. This activity may be applicable to an Air Force level organization. Only at this level could all the information be accumulated and studied. Only then could recommendations be forwarded to the product centers for dissemination among the various programs. A goal to this effect was added.

4.5-2 Use of SAMF. As stated earlier, to be truly effective, the SAMF's use should be limited. The researchers recommend that a group be created at each product center for the purpose of conducting the assessments. The information from the assessments should go only to the program manager and SAPG. It would be the responsibility of the program manager in coordination with the SAPG to analyze the results and take corrective actions.

The researchers noted that no process improvement action can be successful without the full support of senior management. Here, senior management includes the product center commanders, the AFMC commander and the Chief of Staff of the Air Force. However, the program managers also must support the process. Without them, the assessments would not be possible, data would not be available, and improvement may not occur. In short, everyone from the Chief of Staff to the program manager must support the effort for it to be successful.

4.6 Summary

From the research knowledge base developed in Chapter III, the researchers developed a baseline SAMF and a Delphi questionnaire. Air Force software acquisition professionals were surveyed to obtaining additional information and their advise about the acquisition process and the researchers' initial findings. The SAMF was refined based on the survey and literature review results. The Software Acquisition Maturity Framework contains all but one of the CMM's KPAs (modified to suit the acquisition environment) plus several new acquisition specific KPAs.

V. Conclusions

5.1 Introduction

The goal of this research was to develop an Air Force Software Acquisition Maturity Framework based on software acquisition key process areas and the SEI CMM's software development maturity framework. The research was accomplished in three phases. First, Air Force software acquisition key process areas were defined through an in-depth literature/document review. Next, the SEI CMM's framework was adapted to the software acquisition process establishing a baseline SAMF. Finally, Air Force software acquisition experts were surveyed for their advice regarding the software acquisition KPAs and the baseline SAMF. Based on the information gathered, the SAMF was completed. The research results as well as several recommendations are presented in the following sections.

5.2 Research Results

The research revolved around solving four secondary research objectives. By answering the four objectives, the authors provided a solution to the research problem as stated above.

The first objective was to define the Air Force software acquisition process as set forth by applicable DoD and Air Force documents. The DoD 5000 series defines the general acquisition policies and procedures required for defense acquisition management. These policies and procedures apply to all defense programs independent of their acquisition category. However, the literature indicated and the experts confirmed that the DoD 5000 series does not define a standard software acquisition process for the DoD, the Air Force, the AFMC, or the AFMC product centers. Other documents, such as DoD-STD-2167A and DoD-STD-2168, define key parts of the software acquisition process but not the whole process. Instead, each Air Force program must implement the applicable DoD 5000 series policies and procedures to its program and define its own software acquisition process.

What is needed is a standard set of software acquisition processes (elements of the whole process) defined for each application domain at the product center level. Each Air Force program office would then combine any subset of these processes to make a program software acquisition process to meet its needs. This is required because different SPO's use different software acquisition processes depending on their type (i.e. basket SPOs versus major program SPOs), program application (i.e. C³I, versus aircraft), and personnel (i.e. experience level and training). Furthermore, software acquisition management within the SPO should not be segregated to a "Software Directorate." Software acquisition management responsibilities should be delegated to the SPO's project teams. And, there should be at

least one program manager (who is responsible for coordinating the SPO's software acquisition process) to ensure consistency across the program.

The second and third research objectives were to define key process areas within the software acquisition process, organize them, and validate them. Based on the information gathered during the research, the authors concluded that all but one of the CMM KPAs are applicable to the Air Force software acquisition process. The Peer Reviews KPA is not applicable because it is specifically geared for the software development process, therefore it is not germane to software acquisition management. The 17 CMM KPAs are placed in the same maturity levels as in the SEI's CMM. Additionally, four other KPAs, necessary for proper software acquisition management, are defined. Two of these KPA's constitute the initial maturity level. One was placed in the repeatable maturity level, the other in the defined maturity level. These conclusions are based on the opinions and recommendations of software acquisition professionals throughout the Air Force in addition to information gained from the literature reviewed.

The final objective was to develop the SAMF based on the validated KPA list and the SEI's CMM framework. The completed SAMF, illustrated in Figure 15, page 92, has the identical structure as the CMM. Therefore, the five CMM maturity levels and their definitions are adapted to the SAMF without change. Furthermore, the 17 CMM KPAs are modified to reflect software acquisition management characteristics, goals, and commitments to perform, as recommended by the experts. The four additional KPAs are defined based on the literature and the experts' recommendations.

In comparison to the SEI's CMM, The SAMF has two Level 1 key process areas while the CMM has none. This is because the experts indicated that simply following the DoD/Air Force guidance does not guarantee the software acquisition process is anything more than chaotic. Therefore, a Level 1 software acquisition organization must meet the minimum acquisition policies and procedures as set forth by the DoD 5000 series before it can mature to Level 2.

The SAMF defines the AFMC product center as the organization and the SPOs as the projects. This means the product centers should use the SAMF to assess their software acquisition management capability. At the macro-level, weak areas of the product center's process should be improved and strong areas capitalized on. At the micro-level, the SPOs should determine their tailored process's strengths and weaknesses and improve their process accordingly.

Finally and most importantly, the SAMF must not be used as a tool for external evaluations such as those performed by the Air Force inspector general or the General Accounting Office. Use of the SAMF as an evaluation tool will negate the product center's/SPOs' motivation to assess and improve their software acquisition process. Instead, the organization's emphasis will shift from improving their

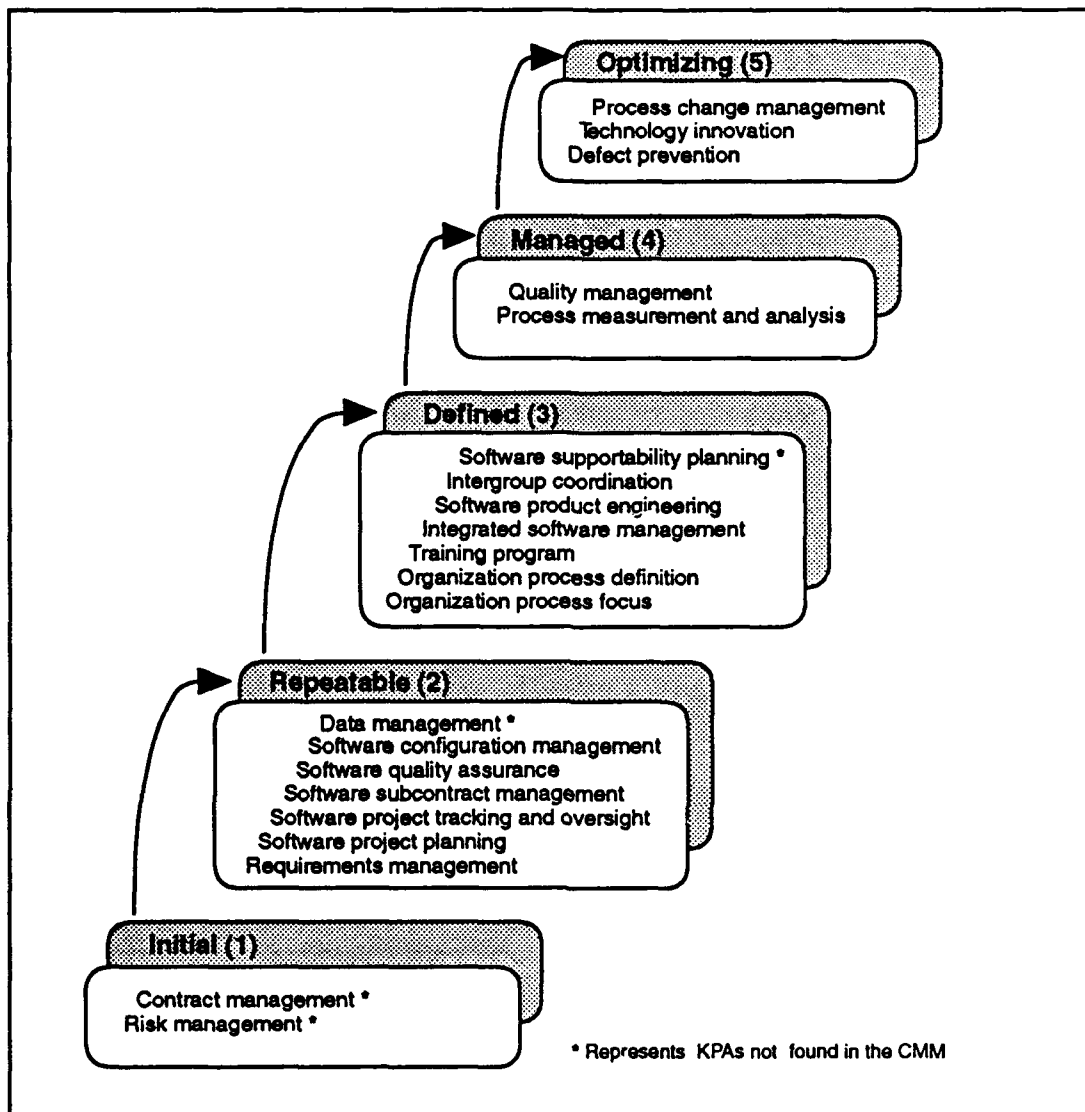


Figure 15. Software Acquisition Maturity Framework
(adapted from Paulk et al, 1991:28)

process to achieving a "better grade." Software acquisition process assessments must be done by the organization and for the organization. This fact cannot be stressed enough.

In answering the four secondary research objectives, the researchers were able to develop a SAMF based on the SEI's CMM. The literature provided information for the selection of important software acquisition areas. The experts and literature provided the definition of the KPAs. The experts and literature validated and organized the list of KPAs. And finally, the experts and the CMM framework

provided the structure of the new SAMF. The Air Force Software Acquisition Maturity Framework is listed in Appendix C.

5.3 Recommendations for Study Replication

The researchers only encountered two major problems in completing this effort: the selection of experts and a high dropout rate among those selected. The researchers recognized that a better approach to the first problem may have prevented the second problem from occurring. However, each problem will be addressed individually.

5.3-1 Location of Experts. Locating individuals that met the expert criteria was indeed a difficult process. The researchers found that several organizations kept lists of individuals who have some expertise in software, but no organization contained all the information necessary. Because of this, the researchers were forced to use personal contacts (mostly through the MCCR Focal Points at the different product centers) to identify the experts and enlist their support. This process proved very time consuming and labor-intensive. Several weeks were spent on phones talking to many different individuals and trying to locate people that met the criteria.

The researchers have two recommendations for the selection process: start early and widen the search. Starting the search early is essential. Many of these candidates work for program offices and travel a great deal. Getting in touch with these individuals was difficult. The researchers believe that there is a direct correlation between the time spent searching and the number of experts enlisted. In support of this, the authors were provided with several lists (some with 50 or more names) to contact as candidates. If time would have permitted contacting all of the listed individuals, several more experts could have been found.

The second means to facilitate the search is to widen the scope. This thesis effort attempted to use candidates from any of the five product centers. However, with the new IWSM approach to systems acquisition, several more experts may reside at the air logistics centers as well. The five ALCs should be included in the search as well. Furthermore, experts who teach for either Air University or Air Training Command could have the experience necessary as well. The researchers did locate individuals at the Systems Acquisition School but TDY schedules precluded their participation. Any future search should included these areas as well.

5.3-2 Dropout Rate of Participants. Of some 20 candidates originally identified, only ten returned both questionnaires. The biggest contributor to this problem was the length of the questionnaires. Both rounds contained over 25 pages of text. By any standard, this was too long.

However, in order to present unbiased questions covering all appropriate topics, the researchers had little choice but to produce an extensive questionnaire. The trade-off between the questionnaire's content validity and the survey's response rate must be carefully considered.

For these reasons, it is recommended that a great many experts be identified and sent both rounds early in the process. By increasing the original numbers of experts, the number that answers both rounds could be increased as well. As a secondary benefit, a broader base of experts might have influenced the results as well. The limited number of experts did not cover all product centers. Furthermore, as the authors found, there were software acquisition experts at places other than the product centers, all of whom could have enlisted in the effort. The identification of more experts would have had a positive impact on the research.

Another recommendation is to design the questionnaire to be more concise. One way this could be accomplished would be to break the original questionnaire into two. One questionnaire would ask the general approach and SAMF application questions, the other would ask the specific key process area questions. The general approach questionnaire should target senior acquisition management while the more specific questionnaire should target lower level management. However, the trade-off mentioned above must be considered carefully if this approach is taken.

5.4 Recommendations for Further Research

From the very start, this thesis effort was recognized as the start of an extended thread of research. The authors intended to develop only the framework that could be expanded into a Software Acquisition Maturity Model by subsequent research efforts. In addition, this effort has brought several issues to light that deserve an entire thesis themselves. The following sections describe the major follow-on efforts recommended by the authors.

5.4-1 Validate Results. As previously mentioned, the group of experts used in this research effort did not represent all AFMC product centers. Furthermore, several locations were found that could provide the needed expertise. Therefore, the authors recommend that a second research effort be undertaken to validate the results with a larger sample of experts representing a greater cross-section of Air Force software experts. If so desired, a validation effort could extend beyond the confines of the Air Force and seek experts from the Navy, the Army, and the Marine Corps.

The main thrust of a validation effort could take the questionnaire used in this thesis and present it to a larger group of experts. Statistical methods could be used on the larger sample to gain more confidence in the central tendencies of the experts. By doing so, the results gained in this thesis effort could be validated or proven non-representative. If a larger population was selected, the framework

could be validated for the DoD as a whole. Whatever the scope, a validation effort is the next logical step in the research thread.

5.4-2 Expand the SAMF into a Software Acquisition Maturity Model.

Once the results are validated, research could focus on expanding the framework into a full maturity model. The SAMF only provides goals and commitments for each of the KPAs. The CMM, a full maturity model, provides additional information in the form of key practices and key indicators. The key practices and indicators identify the organization's abilities and activities to perform the KPA as well as the steps taken to monitor and verify proper implementation. Research could be undertaken to expand the SAMF to provide this type or similar information. The research objective of such an effort could be to develop, from the SAMF, a Software Acquisition Maturity Model (SAMM).

5.4-3 Develop an Assessment Methodology and Guidance.

The CMM uses the activities and the monitoring and verifying functions as the basis for the questions used in the software process assessment and software capability evaluation methodologies. Therefore, once a SAMM is developed, the next logical step is to provide the implementing methodology. The authors point out that a methodology currently exists in the form of the SEI's software process assessment. Any research in this area should start there. However, since the underlying content of the CMM is different than the SAMF, a new acquisition process assessment questionnaire should be developed. The research effort could focus on developing the questionnaire and possibly attempt to test its implementation. The goal of this research effort could be to develop the implementation tool and methodology for applying the SAMM for software acquisition process improvement.

5.4-4 Develop Process Improvement Methodologies and Guidance.

Once a tool and its methodology have been developed, the SAMM could become useful to the Air Force or the entire DoD. However, simply being able to determine an organization's software acquisition maturity level is insufficient. The organization must have guidance for improving and maturing its process. This guidance must address maturing from any given level to the next. It also must be generic enough to be used by any organization that procures custom software. Possible organizations include product centers, ALCs, and major command headquarters, to name a few. The research purpose could be to provide process improvement guidance for organizations wishing to improve their software acquisition process.

5.4-5 Research the Effects of the IWSM Approach to the Framework or Model.

With the joining of Systems Command and Logistics Command, the Air Force is

adopting an Integrated Weapon Systems Management approach to its acquisition and support process. Under this program, a single program office (with individuals at both the product center and the ALC) will be responsible for the procurement and life-cycle support of a given system. This cradle-to-grave approach may indeed have an impact on the structure and content of the SAMF or any subsequent SAMM. Currently, the IWSM process is too immature to assess any possible impacts. However, in the years to come, the process may become deeply ingrained into the Air Force acquisition doctrine. At that time, a research effort could be undertaken to refine or adapt the SAMF or SAMM to represent the key process areas of a software acquisition effort under IWSM.

5.4-6 Develop a Training Program Template. One of the key process areas identified in the SAMF is a training program. The goals of this program include: training the staff to understand the organization's software acquisition process and to effectively use the capabilities and features of the existing and planned organization work environment; providing the staff with opportunities to improve their professional skills; and ensuring the organization effectively manages the turnover of personnel and the training of new individuals in the organization's software acquisition process. These goals are high-level in nature and do not describe or indicate what type of training is necessary for software acquisition professionals. A possible research effort could attempt to determine what type of training is necessary for these software individuals. Furthermore, the research could indicate where a program office could send its personnel to obtain this training. This effort need not wait until a Software Acquisition Maturity Model is developed. Actually, Space and Missile Center has already voiced an interest in this topic and is willing to support such an effort.

5.4-7 Develop a Software Metrics Program Template. The SAMF indicates that a metrics program should be undertaken to measure the acquisition process. While many software development metrics are available, software acquisition metrics have not yet appeared. Therefore, an entire stream of research could be started to develop acquisition metrics that can be used to quantitatively measure the software acquisition process. This stream of research could involve several theses as it may migrate into the systems acquisition world.

A second metric research topic could be guidance for the selection of a developmental metrics set to be require of the contractor. The organization could use the guidance to setup its own developmental metrics program to feed its acquisition process measurement. The purpose could be to develop a methodology for selecting a set of software metrics appropriate for a given organization.

5.4-8 Produce a Development/Acquisition Maturity Matrix. In the early stages of this research effort, the authors interviewed Mr. Woody Mead, a member of the SEI's technical staff (Mead, 1991a). Mr. Mead brought up the idea of developing a matrix of the contractor's developmental maturity versus the government's acquisition maturity. This matrix itself would be very easy to build. However, to be useful, the product must also provide management guidance to the government. This guidance could include general guidelines for managing the contractual effort. For example, if the contractor is a Level 4 software developer and the government organization is a Level 1 software acquirer, a hands-off management approach might be in order. However, if the position were reversed, the government may wish to get heavily involved in the developmental process. The research objective could be to provide the government managerial guidance to the program manager based on both party's process maturity.

5.4-9 Determine if the Higher Levels of the Framework Should Apply to the General Acquisition Process. One of the experts polled in this thesis stated that the higher maturity levels should de-emphasize software acquisition and focus on system acquisition. It was stated that the government generally procures an entire system; software is usually only a portion of the system. Therefore, it would seem reasonable that any acquisition maturity model would take this into account. The expert noted that some of the Level 3 and all of the Level 4 and 5 activities were not software specific and could be applied to the entire acquisition process. The reasoning suggesting that a more mature organization would place less emphasis on any one portion and more on the entire system. Research could be undertaken to support this position.

5.4-10 Develop an Acquisition Process Maturity Model. Research could also be undertaken to develop a general acquisition maturity model in much the same fashion that this stream of research was started. A second expert noted that an acquisition maturity model should be developed—an alternate approach was suggested, however. The expert suggested any such research should not attempt to modify the KPAs of the CMM but rather use a top-down approach. Professional acquisition personnel could be interviewed to determine what the key practices of systems acquisition are. From there, a framework could be developed and an implementation tool produced. Again, this could be the beginning of an entirely separate thread of research.

5.4-11 Other Topics. The previous sections described several topics that can be researched as follow-up efforts to this thesis. Furthermore, several separate research streams were

identified as well. One was an evaluation/comparison of the SEI's CMM and ASC's Software Development Capability Capacity Review (SDCCR).

As noted in Chapter III, the SEI's CMM was developed under contract to Electronic Systems Center. However, during the same time the CMM was being developed, the Aeronautical Systems Center developed a separate methodology to evaluate a contractor's software development capabilities. This methodology is called the Software Development Capability Capacity Review. ASC has published a pamphlet on its use and has used the methodology on a few source selections including the Advanced Tactical Fighter program.

There has been discussion within AFMC to combine the best parts of the two methodologies and develop an AFMC standard software capability evaluation tool. This tool could be used for source selections and process improvement incentive programs. To the authors' knowledge, little if any action has been taken to meld the two methodologies.

Multiple thesis efforts could be derived from an evaluation of the two methodologies. For example, one thesis could evaluate and compare the results achieved in the use of these two methodologies for source selection. Another could perform a similar evaluation for their use in government incentive programs aimed at improving the contractor's software development capabilities. Yet a third line of research could attempt to combine the best parts of both methodologies to develop an AFMC standard software development capability evaluation tool.

In short, this thesis effort opened up the possibilities of several other research efforts. The possible topics covered here were not meant to be all-inclusive. The authors realized that there may be several other research topics generated by this thesis that were not mentioned. The point is clear: more research is needed in the area of software acquisition process improvement.

5.5 Summary

The purpose of this research was to develop an Air Force Software Acquisition Maturity Framework based on the key processes of the Air Force software acquisition process and the software development maturity framework found in the SEI's CMM. In doing so, the authors reviewed pertinent literature in an attempt to define a standard Air Force process. This, however, was not possible. From the review, though, several important processes were identified. Experts in software acquisition were sought and surveyed via a two-round Delphi questionnaire. The information gathered from the questionnaires and the literature review was used to define, order, and validate the list of Air Force software acquisition key process areas. The researchers then used information from the experts and the literature to adapt

the CMM structure and integrate the KPAs. The final result of this effort was the Air Force Software Acquisition Maturity Framework (SAMF) documented in Appendix C.

Appendix A

Delphi Questionnaire

This questionnaire is designed to elicit your opinions and recommendations on the proposed adaptation of the Software Engineering Institute's (SEI) Capability Maturity Model (CMM) to the Air Force Systems Program Offices' (SPO) software acquisition process. The primary focus of this questionnaire is to compare the SEI's CMM to the software acquisition process. However, the questions are not intended to be exhaustive.

Straight-forward, candid responses are essential to the research. Comments, opinions and

recommendations are highly encouraged. Without them, this survey has little value. "No Opinion" is not part of the rating scale, nor does it imply neutrality. Rather, "No Opinion" indicates you have no opinion about the statement.

Two rounds of questioning will be conducted. We will provide feedback on the first round of the survey by compiling, summarizing, and incorporating round one responses into the round two questionnaire.

Instructions

- ❶ Please check the box best corresponding to your degree of agreement for each question.

Please write any comments to include the following as applicable:

- a. reasons supporting your choices.

b. recommendations or ideas concerning the adaptation of the SEI's CMM to the Air Force software acquisition process.

- ❷ Please make your comments and recommendations in the space provided or on the back of the questionnaire sheets, numbering these as needed.

Privacy Act Statement

According to paragraph 8, AFR 12-35, the following information is provided in accordance with the Privacy Act of 1974:

a. Authority:

- (1) 5 U.S.C. 301, Departmental Regulations, and/or
- (2) 10 U.S.C. 8012, Secretary of the Air Force, Powers, Duties, Delegation by Compensation, and/or
- (3) DOD Instruction 1100.13, 17 Apr 68, Surveys of Department of Defense Personnel, and/or
- (4) AFR 30-23, 22 Sep 76, Air Force Personnel Survey Program.

b. Principal Purpose. This survey is being conducted to collect information to be used in research aimed at illuminating and providing inputs to the solution of problems of interest to the Air Force and/or DOD.

c. Routine Uses. The survey data will be converted to information for use in research of management related problems. Results of the research, based on data provided, will be included in a master's thesis and may also be included in published articles, reports, or texts. Distribution of the results of the research, based on the survey data, whether in written form or presented orally, will be unlimited.

d. Participation in this survey is entirely voluntary.

Research Objectives

The objective of this research is to adapt the SEI's Capability Maturity Model (CMM) to the Air Force software acquisition process creating a Software Acquisition Maturity Framework (SAMF). For this research, the software acquisition process is defined as the management process used by the System Program Offices (SPO) within the Air Force Material Command (AFMC) Product Centers to procure software. The proposed SAMF would *not* be used to measure the SPO's

software *development* capability but to measure the SPO's software *acquisition management* capability. Hence, a user of the proposed SAMF views software development from a much different perspective than those using the SEI's CMM. This questionnaire is designed to ascertain your opinions and recommendations as to the feasibility and specific attributes the proposed SAMF should contain.

Questionnaire Assumptions and Limitations

It is assumed you understand the structure, content, and objectives of the SEI's CMM. If this is not the case, answering many of the questions will be very difficult. Please reference SEI documents SEI-91-TR-24 [*Capa-*

bility Maturity Model for Software] and SEI-91-TR-25 [*Key Practices of the Capability Maturity Model*] for specific information about the SEI's CMM including definitions not annotated within this document.

Summary of the SEI's CMM

This summary is based on the SEI's CMM description in SEI-91-TR-24. The CMM is designed as a tool for measuring an organization's software development process maturity. The model is founded on the premise that the process maturity level indicates an organization's software development capability. The structure of the CMM is defined by a framework of five maturity levels. These maturity levels define an ordinal scale by which process maturity can be defined, measured, and evaluated. Definition of these levels

was based on previous work by Philip Crosby [*Quality is Free*, 1979] and Ron Radice ["A Programming Process Study," *IBM Systems Journal* Vol. 24, No. 2, 1985]. Each maturity level has a corresponding set of Key Process Areas (KPA) defining practices/characteristics a software development organization should have to achieve that maturity level. Assessing which KPAs (if any) an organization meets defines that organization's maturity level, thereby indicating the organization's software development capability.

Premise of Research

The premise of this research is that the same "benchmarking" approach of measuring an organization's software development process

can be used to measure the Air Force System Program Offices' process of managing software acquisition. The proposed SAMF looks

at software development from the acquisition management's point of view, while the CMM looks at software development from the developer's point of view. While the CMM is constructed to be used as a tool for both Software Capability Evaluations (used by the government acquisition community to evalu-

ate contractor's software development capabilities), and Software Process Assessments (used by the software development organization as a self assessment tool), *the proposed SAMF is targeted strictly for a SPO self assessment capability.*

Outline of Questionnaire

The questionnaire is broken into six (6) sections. The initial section requests data on yourself. The second section ascertains your initial reaction to the proposed SAMF. The third section establishes your opinion on the organizational level a SAMF should be targeted for within the Air Force software acquisition process. The fourth, fifth, and sixth sections request opinions and recommendations on adapting the SEI's CMM to the software acquisition management process.

One of the difficulties in implementing this questionnaire was the researchers had to define the context in which the questions should be answered. This meant some assumptions were made about the respondent's "point of view" on the proposed SAMF. We request you answer the questions in the fourth, fifth, and sixth sections to the best of your ability and please keep in mind the context in which they are asked.

Demographics

- Please write your name.
- What is your grade/rank?
O-__
GS-__
GM-__
- What is your current assignment and job title?
- How many years experience do you have in Air Force acquisition?
- How many years of the acquisition experience are specifically related to software (i.e. held position as MCCR focal point for program Test, Engineering, Projects, Quality Assurance or any of the other functional disciplines)?
- There are many functional disciplines in a SPO organization. Please indicate which of the following disciplines you have experience in (please check all applicable).
 - ☐ Project Management
 - ☐ Software Engineering
 - ☐ Contracting
 - ☐ Logistics
 - ☐ Software Test & Evaluation
 - ☐ Software Quality Assurance
 - ☐ Software Configuration Management
 - ☐ Other(s) _____
 - _____
 - _____
 - _____

- Which acquisition organizations have you been assigned to (please check all applicable)?

- ☐ ASD
- ☐ ASD-South (MSD)
- ☐ BMO
- ☐ ESD
- ☐ HSD
- ☐ SSD
- ☐ Other _____
- _____
- _____

- Have you been trained on the SEI's CMM or are you familiar with it (please check all applicable)?

- ☐ Software Capability Evaluation (SCE) training at the SEI
- ☐ Software Process Assessment (SPA) training at the SEI
- ☐ Executive SCE/SPA training at the SEI
- ☐ In house (SPO/office) training
- ☐ Software Professional Development Program (SPDP)
- ☐ Computer Resource Acquisition Course (CRAC)
- ☐ Personal training (i.e. read Humphrey's "Managing the Software Process")
- ☐ Other _____
- _____
- _____

General CMM/SAMF Questions

The following questions are designed to ascertain your initial reaction and overall view of the proposed SAMF.

- ➡ **1. The SEI's CMM is a valid tool for contractors to self assess and improve their software development process.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

- ➡ **2. Do you think a software acquisition organization within a SPO could use a tool such as the proposed SAMF to benchmark or measure their management process for self improvement?**

☐ Yes ☐ No ☐ No Opinion

➡ Please list your reasoning (i.e. the pros and cons) of why or why not:

III ➡ 3. Do you think the idea of adapting the SEI's CMM to the Air Force software acquisition process is feasible?

☐ Yes ☐ No

☐ No Opinion

➡ Please list your reasoning (i.e. the pros and cons) of why or why not:

III ➡ 4. To be able to benchmark or measure a process one has to be able to define the process. The Air Force software acquisition process can be defined from current literature such as DoD directives and Air Force policy.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

III ➡ 5. There is a "standard" Air Force software acquisition process used at all Air Force Material Command (AFMC) product centers.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

III ➡ 6. The Air Force software acquisition process is the same for all program levels (i.e. basket SPOs versus major program SPOs).

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

III ➡ 7. The management of software acquisition can be defined as a stand alone functional discipline within a SPO.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

Target Air Force Organization for SAMF Tool

The research objective is to ascertain the feasibility of adapting the SEI's CMM to the Air Force SPO software acquisition process. However, we would like your opinions/recommendations as to the correct "level of focus" you feel a tool such as the proposed SAMF should target. Therefore, please answer the following questions *after reading* key points taken from SEI-91-TR-24 and SEI-91-TR-25.

CMM Software process assessments are targeted at the software development *organization* level. SEI-91-TR-25 defines an organization as "a unit within a company,

agency, or service that shares common management, is centered at a single geographical site, and has responsibility for a common business area." The scope of the software development organization may differ depending on the company size and assessment strategy (i.e. evaluating the maturity level of a company's division versus a unit within the division). Furthermore, the methodology for implementing the maturity assessment implies *multiple projects* should be evaluated to determine the *organization's* process maturity.

➡ **8. Considering the key points listed above, how should the CMM organization/project assessment focus be mapped to the Air Force software acquisition management process?**

- ☐ Target the SAMF process self assessment tool at the SPO level where the SPO is the organization and the segments/sub-systems/teams are the projects.
- ☐ Target the SAMF process self assessment tool at the SPO level where the SPO is both the organization and the project.
- ☐ Target the SAMF process self assessment tool at the AFMC product center level where the product center is the organization and the SPOs are the projects.
- ☐ Target the SAMF process self assessment tool at the Program Executive Officer (PEO) level where each PEO program grouping is the organization and the SPOs are the projects.
- ☐ Other recommended mapping _____

➡ Please list your reasoning (i.e. the pros and cons) of why or why not:

Definition of Acquisition Environment Context

The remaining questions require the "context" which you should assume when answering the questions. The problem with defining the "context" is that it may be opposite to your opinions stated in the previous questionnaire sections. Please keep in mind the research objective—namely to adapt the SEI's CMM to the SPO software acquisition management process. Therefore, we request you answer the following sections to the best of your ability in reference to the "context" defined below.

When answering the following questions please assume:

- Adapting the SEI's CMM to the software acquisition process is a valid approach for
- developing a self assessment tool to measure the SPO's software acquisition process maturity.
- The SAMF process self assessment tool is targeted at the SPO level where the SPO is both the "organization" and the "project."
- The SAMF tool is designed to assess only the SPO's *software* acquisition management capabilities.
- The program manager has have overall management authority over the Mission Critical Computer Resources (MCCR) portion of a major systems acquisition during the Engineering and Manufacturing Development (EMD) phase.
- The SPO is located at an AFMC Product Center.

CMM Maturity Level Mapping to SAMF

The SEI's CMM is based on a framework of five maturity levels. The maturity levels are defined to be consistent with the premise that process maturity indicates software development capability. This section's objective is to elicit your opinions and recommendations as to the applicability of the SEI's CMM maturity levels to the Air Force software acquisition process. Each CMM maturity level is listed

below with its corresponding definition from pages 8 and 9 of SEI-91-TR-24. Please keep in mind the context defined above, and answer the question(s) which follow(s) each maturity level.

(1) Initial—The software process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort.

*** From the software acquisition process point of view:**

➡ **9. Is this maturity level applicable to the proposed SAMF?**

☐ Yes ☐ No

☐ No Opinion

10. *The Air Force software acquisition process is based on existing directives and regulations (i.e. DoD 5000.1 & DoD 5000.2). However, even when these "minimum" management processes are implemented, the software acquisition process can still be ad hoc.*

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

(2) **Repeatable**—Basic project management processes are established to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.

** From the software acquisition process point of view:*

11. *Is this maturity level applicable to the proposed SAMF?*

☐ Yes ☐ No ☐ No Opinion

12. *This maturity level implies a "repeatable software acquisition project management" capability. However, there is nominally only one "project" within a SPO, hence, the capability to repeat the procurement of a similar project is not an important attribute of the SPO.*

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

13. *If this maturity level is not applicable, the KPAs from this level should be integrated into another maturity level.*

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

(3) Defined—The software process for both management and engineering activities is documented, standardized, and integrated into an organization-wide software process.

All projects use a documented and approved version of the organization's process for developing and maintaining software.

*** From the software acquisition process point of view:**

14. Is this maturity level applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

15. The characteristics of tracking cost, schedule, and functionality (originally in the repeatable level) should be integrated into the defined level.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

(4) Managed—Detailed measures of the software process and product quality are collected. Both the software process and

products are quantitatively understood and controlled using detailed measures.

*** From the software acquisition process point of view:**

16. Is this maturity level applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

17. This maturity level implies qualitative data is gathered on the software acquisition process.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

(5) **Optimizing**—Continuous process improvement is enabled by quantitative feedback from the process and from testing innovative ideas and technologies.

** From the software acquisition process point of view:*

➡ **18. Is this maturity level applicable to the proposed SAMF?**

☐ Yes ☐ No

☐ No Opinion

➡ **19. This level implies quantitative data is taken on the process. Gathering quantitative data takes time, however, with the new Integrated Weapons Systems Management acquisition process, opportunities for SPOs to take quantitative data will exist.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

CMM Key Process Area Mapping to SAMF

The SEI's CMM is based on a framework of five maturity levels. Each maturity level is composed of a set of KPAs which define practices and characteristics an organization should have to achieve that maturity level. This section's objective is to elicit your opinions and recommendations as to the applicability of the CMM Key Process Areas to the Air Force software acquisition process. Each Key Process Area listed below is as stated in SEI-92-TR-25. The questions that follow request your opinions and recommendations on the applicability of each CMM KPA to the software acquisition management process.

Please keep in mind the context defined above when answering the questions.

Requirements Management involves establishing and maintaining an understanding and agreement with the customer on the requirements for the software throughout the software's life cycle. The agreements cover both the technical requirements for the software and the nontechnical requirements, such as delivery dates for the software. The agreements form the basis for estimating, planning, performing, and tracking the project's software activities.

** From the software acquisition process point of view:*

➡ **20. Is this Key Process Area applicable to the proposed SAMF?**

☐ Yes ☐ No

☐ No Opinion

➡ **21. Requirements Management involves establishing a working relation with the "user" (customer). This includes understanding and planning for the implementation of "user" requirements.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

➡ **22. Requirements Management involves establishing a working relation with the "contractor" (seller). This includes contracting for and maintaining correct allocation of "user" requirements.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Software Project Planning involves developing estimates for the work to be performed, establishing the necessary commitments, and defining the plan to perform the work. A plan is established to address the commitments to

the customer according to the resources, constraints, and capabilities of the project. The plan provides the basis for initiating the software effort and managing the progress of the work.

** From the software acquisition process point of view:*

➡ **23. Is this Key Process Area applicable to the proposed SAMF?**

☐ Yes ☐ No ☐ No Opinion

➡ **24. Software Project Planning involves ensuring the contractor's software development plan is adequate and feasible.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

➡ **25. Software Project Planning involves addressing the government activities necessary to implement the contractor software development plan (e.g. a detailed plan explaining the government technical review process of deliverables).**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Software Project Tracking and Oversight involves tracking and reviewing the software accomplishments and results against documented estimates, commitments, and plans, and adjusting these based on the actual accomplishments and results. A documented plan for the software effort is used as the basis for tracking the software activities, commu-

nicating status, and revising plans. The software activities are monitored by the software managers on a regular basis. Regular technical reviews and reviews with the project manager and senior management are conducted to ensure that management and staff are aware of the software status and plans, and that issues receive appropriate attention.

*** From the software acquisition process point of view:**

26. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

27. Software Project Tracking and Oversight involves tracking and reviewing the accomplishments of the contractor in accordance with the contract.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

28. Software Project Tracking and Oversight involves tracking and assessing the software acquisition management process in accordance to the Software Project Plan.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

Software Subcontract Management involves selecting a software subcontractor, establishing commitments with the subcontractor on the work to be performed, coordinating activities with the subcontractor, and tracking and reviewing the subcontractor's performance and results. The subcontractor is selected based on their ability to perform the work. A documented agreement covering the

technical and nontechnical (e.g., legal, financial, and administrative) requirements is established and is the basis for managing the subcontract. Regular technical and management reviews are conducted to ensure that management and staff of both organizations are aware of the software status and plans, and that issues receive appropriate attention.

*** From the software acquisition process point of view:**

29. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

30. Subcontract Management is a function of the Prime Contractor. This should not be management directly by the SPO. Instead, subcontractor management requirements should be stated clearly in the prime contract.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

Software Quality Assurance involves reviewing and auditing the software products and activities to ensure that they comply with the applicable processes, standards, and procedures, and provides the staff and managers with the results of their reviews and audits. The software quality assurance function is required on all projects. The group performing this function is independent of the soft-

ware groups and project management. A senior manager who is committed to handling all major software quality assurance issues is identified. Where compliance issues exist, the software quality assurance group works with the appropriate managers, including senior management where required, to resolve the issues.

*** From the software acquisition process point of view:**

31. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

32. Software Quality Assurance is a function of the contractor's software development organization and should not be managed by the SPO as a separate distinct discipline.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

33. The function of the SPO's Software Project Management is to ensure that the SQA function of the contractor is properly performed.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Software Configuration Management involves selecting project baseline items (e.g., the project description, products, and process specifications of the project), controlling these items and changes to them, and recording and reporting status and change activity for these items. Changes to these baseline

items are controlled systematically using a defined change control process. The configuration (software and documentation) of a system, or of any of the controlled intermediate or support products, can be distinctly identified at any point in time.

*** From the software acquisition process point of view:**

34. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No ☐ No Opinion

35. Software Configuration Management is a function of the contractor's software development organization and should not be managed by the SPO as a separate discipline.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

- III ➡ **36. The function of the SPO Configuration Management Directorate, in close coordination with the project and engineering directorates, is to ensure the SCM function of the contractor is properly performed.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Organization Process Focus involves developing and maintaining an understanding of the organization's software processes and coordinating the activities to specify and improve these processes. A group such as a software engineering process group acts as the

focus for the software process activities in the organization. This group coordinates the projects' software process definition activities and the organization's long-term process improvement efforts.

*** From the software acquisition process point of view:**

- III ➡ **37. Is this Key Process Area applicable to the proposed SAMF?**

☐ Yes ☐ No ☐ No Opinion

- III ➡ **38. Organizational Process Focus implies a software acquisition process group acts as the focal point for process improvement. However, implementation of such a group at the SPO level is not feasible.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

- III ➡ **39. A software acquisition process Group (SAPG) would best be implemented at the AFMC product center level.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Organization Process Definition involves establishing and maintaining a standard software process for the organization, along with related items, for use by the projects in establishing their software process. This standard software process provides a common process

for software development and software maintenance projects. It defines the essential process steps for the projects and establishes the common basis for measuring process performance across all projects and for long-term improvement.

*** From the software acquisition process point of view:**

40. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

41. Organizational Process Definition implies all product center SPOs implement a standard software acquisition process.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

Training Program involves identifying the training needs of the organization, the projects, and the individuals and developing, and procuring training courses to address these

needs. The training program ensures that training needed to perform each of the organization's job functions is appropriate and is not circumvented inappropriately.

*** From the software acquisition process point of view:**

42. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

43. A SPO Training Program should address the organization's general acquisition training requirements (e.g. training needs to meet the Acquisition Professional Development Program requirements).

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

- ➡ **44. A SPO Training Program should address the training needs of the individuals managing the software acquisition process oriented to the specific procurement being performed (e.g. training individuals on the user requirements for the project).**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Integrated Software Management involves establishing and maintaining the project's defined software process and managing the software activities according to this defined software process and the special needs of the project. The project's defined software process integrates the management and technical processes as the basis for performing the pro-

ject's activities. When project objectives are not being achieved, the managers know what actions need to be taken to correct the problem and reduce the likelihood of similar problems in the future. The organization provides support and historical data that the project uses to improve its software estimating, planning, and tracking process.

*** From the software acquisition process point of view:**

- ➡ **45. Is this Key Process Area applicable to the proposed SAMF?**

☐ Yes ☐ No ☐ No Opinion

- ➡ **46. Integrated Software Management implies a standard software acquisition process is defined at a higher organizational level than the SPO (i.e. product center).**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Software Product Engineering involves performing the technical activities to build and maintain the software system using appropriate state-of-the-practice tools and methods. The software requirements are identified, analyzed, refined, and documented. A software architecture and software designs are developed to implement the software requirements. The program code is developed to

implement the software architecture and software designs. The software evaluation activities ensure that the software product to be delivered satisfies the specified requirements. A balance of flexibility and control is maintained to correct and revise the requirements, designs, and code to incorporate corrections and enhancements identified throughout the life cycle.

*** From the software acquisition process point of view:**

47. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

48. Software Product Engineering is a function of the contractor software development organization and not a function of the SPO.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

Intergroup Coordination involves the disciplined interaction and coordination of the project groups with each other to address system-level issues and activities. System-level objectives and plans are established and used as the cornerstone for all project activities. The project groups participate, as appropriate, in defining the system requirements; establishing a system configuration; allocating

requirements to hardware, software, firmware, and manual processes; monitoring and reviewing the hardware and software design and development; and managing and controlling changes to the system throughout the development effort. The technical working interfaces and interactions between groups are planned and managed to ensure the quality and integrity of the entire system.

*** From the software acquisition process point of view:**

49. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

➡ **50. Intergroup Coordination is a very important attribute of both the software development organization and the SPO software acquisition management organization.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Peer Reviews involve a methodical examination of work products by the producer's peers to identify defects and areas where changes and improvements are needed. The specific work products that will undergo peer review are identified as part of the software planning activities. Peer reviews follow defined proce-

dures. These procedures cover: preparing for the review, conducting the review, reporting the results of the review, and certifying review readiness/completion criteria. Problems identified in the review findings are documented and tracked until they are resolved.

** From the software acquisition process point of view:*

➡ **51. Is this Key Process Area applicable to the proposed SAMF?**

☐ Yes ☐ No ☐ No Opinion

➡ **52. Peer Reviews are a function of the contractor software development organization. The use of Peer Reviews is a decision made by the contractor and is not affected by the SPO software acquisition process.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Process Measurement and Analysis involves taking performance measurements of the organization's standard software process (as instantiated by the projects), analyzing these measurements, and making adjustments to

stabilize the process performance within acceptable limits. The process and the associated measurements are established as a baseline and are used to plan and control the process in quantitative terms.

*** From the software acquisition process point of view:**

III ➡ 53. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

III ➡ 54. Process Measurement and Analysis is an attribute a SPO software acquisition organization should have at a mature level. The philosophy of this attribute holds the same for both the software acquisition process and the software development process.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

Quality Management involves defining software quality goals, establishing plans to achieve these goals, and monitoring and adjusting the software plans, activities, and quality goals to improve customer and end-user satisfaction. Quantitative product quality goals are established based on the needs of the organization, customer, and end-users.

Plans and process quality goals are established and the project's software process is specifically adjusted to achieve the product quality goals. The software activities and results are assessed against the quality objectives on a regular basis, and corrective actions are taken to bring forecasted process and product quality in line with the goals.

*** From the software acquisition process point of view:**

III ➡ 55. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

- ➡ **56. Quality Management implies quality software product and acquisition process goals are established and tracked. The philosophy of this attribute holds the same for both the software acquisition process and the software development process.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Defect Prevention involves analyzing defects that were encountered in the past and taking action to prevent the injection of these types of defects in current and future project activities. Software activities are systematically reviewed by those who perform them to identify the defects that were encountered, to

understand the root causes of the defects, and to determine the implications of the defects on future activities. Trends are analyzed to determine the kinds of defects that were encountered in the past. Defects that are likely to recur are identified and specific actions are taken to prevent them.

*** From the software acquisition process point of view:**

- ➡ **57. Is this Key Process Area applicable to the proposed SAMF?**

☐ Yes ☐ No ☐ No Opinion

- ➡ **58. Defect Prevention implies knowledge of when, where, and how defects are injected in the software product or process (either development or acquisition process). Furthermore, steps are taken to prevent similar defects from appearing again. The philosophy of this attribute holds the same for both the software acquisition process and the software development process.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Technology Innovation involves identifying, selecting, and evaluating new technologies, and incorporating the appropriate technologies into the organization's processes. A group acts as the focus for introducing technology innovations into the organization. By maintaining an awareness of software technology innovations throughout the world and systematically evaluating and experimenting

with them, the organization selects appropriate technologies to improve its productivity and product quality. Pilot efforts are performed to assess new and unproven technologies before they are introduced across the organization. With appropriate sponsorship of the organization's management, the selected technologies are incorporated into the organization's process.

➡ ***From the software acquisition process point of view:***

➡ ***59. Is this Key Process Area applicable to the proposed SAMF?***

☐ ***Yes*** ☐ ***No***

☐ ***No Opinion***

➡ ***60. Technology Innovation implies a group is responsible for identifying, selecting, and evaluating new technologies for possible implementation into the acquisition management process. The implementation of such a group should be performed at a higher management level than the SPO (i.e. the product center).***

☐ ***Disagree Strongly*** ☐ ***Disagree*** ☐ ***Agree Strongly*** ☐ ***Agree***

☐ ***No Opinion***

➡ ***61. Technology Innovation at the SPO level involves exploiting new technologies identified by the evaluation group to improve the software acquisition management process.***

☐ ***Disagree Strongly*** ☐ ***Disagree*** ☐ ***Agree Strongly*** ☐ ***Agree***

☐ ***No Opinion***

Process Change Management involves defining process improvement goals and systematically identifying, evaluating, and implementing improvements to the organization's standard software process and the projects' defined software processes on a continuous basis. Appropriate training and incentive programs are established to allow

and encourage all staff and managers to participate in these process improvement activities. Improvement opportunities are identified and evaluated for potential pay-back for the organization. Pilot efforts are performed to assess new and unproven process changes before they are introduced across the organization.

*** From the software acquisition process point of view:**

62. Is this Key Process Area applicable to the proposed SAMF?

☐ Yes ☐ No

☐ No Opinion

63. Making changes for improvement to the software acquisition management process must be planned, implemented carefully, and measured to verify improvements were made.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

Additional SAMF Key Process Areas NOT Defined in CMM

The objective of this section is to elicit your opinions and recommendations as to other Key Processes Areas which should be defined in the proposed SAMF. Please feel free to list

any other practices/characteristics you think a software acquisition organization should have.

64. Risk Management is required by DoD 5000.1 and 5000.2. It is an integral part of the overall SPO acquisition management scheme. The characteristics of identifying and managing software cost, schedule, and performance risks should be a KPA in the proposed SAMF.

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree

☐ No Opinion

- ➡ **65. Contract Management is a key function of any acquisition organization. This is the function of ensuring the contract meets both the user, maintainer, and SPO acquisition requirements. The characteristic of timely and efficient contract modifications should be a KPA in the proposed SAMF.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

- ➡ **66. Data Management entails ensuring the correct software documentation is acquired from the contractor who meets the requirements of the SPO, the user, and the maintainer. Furthermore, the amount of data should be periodically evaluated to ensure only the required amount is procured. The characteristic of accurate and efficient data management should be a KPA in the proposed SAMF.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

- ➡ **67. Software Supportability Planning is a key function of the software acquisition management process. Software support has been stated to account for approximately 60%-80% of the software life cycle costs. Furthermore, the decisions made during the early phases of the system life cycle have a great affect on the supportability of the software. The characteristics of continual software supportability planning throughout the software life cycle should be a KPA in the proposed SAMF.**

☐ Disagree Strongly ☐ Disagree ☐ Agree Strongly ☐ Agree ☐ No Opinion

Thank you for completing this questionnaire.
Should the return envelope be missing or lost,
please return your questionnaire to: (Address).

Appendix B

This appendix contains a complete listing of the Delphi survey results. All expert comments and the group opinion are listed for each question. The comments are segregated into their respective rounds. Each comment has the associated expert response stated. Finally, the "group opinion" represents the results of the entire group, not just those experts who stated comments.

Comments on Overall Impression of the SAMF

• Round I

Very first impression is contained in last line of page 2 of 22 ("the proposed SAMF is targeted strictly for a SPO self assessment capability"). I strongly feel that this is a very good approach (and perhaps essential). With the acceptance of TQM by the Air Force and DoD, this fits in real well. The concern is that it may be used by external staffs (e.g., MAJCOM staff assistance visits, IGs) either directly (where they use the questionnaire) or indirectly (where they use the SPO's answers or how SPO uses) as basis for their evaluations. This may cast a stigma or create a reluctance to use this tool. (I realize that this is the anti-thesis of both TQM and the SEI's CMM philosophies, but as you probably realize, it is still a cultural condition that will need to be dealt with.)

• Round II

The research area and methods are critical to the future of national defense and the Air Force weapons systems acquisition process. This individual believes

your research is vital to both effective, controlled change and the "program management" or "project engineering discipline."

A. "Benchmarking:" by definition of the origination of the term is limited to the individual craftsman's bench and supports repeatability for an individual. The objective seems to be a quest for application of a "universal" measurement standard for software acquisition.

1. Premise of Research—A cogent coherent comprehensive definition of the component areas of "Software Acquisition Management Process" is not asserted as a basis for the research. My understanding of graduate level research, below the doctoral level, is normally based on existing, given, foundation information with the appropriate citations. This does not appear to be the case in your research, or at least is not demonstrated in the survey.

2. The presentation of the elements separately obfuscates the lack of flexibility that would result.

CMM/SAMF Questions and Comments

1. The SEI's CMM is a valid tool for contractors to self assess and improve their software development process.

• Round I—Group Opinion: Agree

[AGREE STRONGLY] We have seen dramatic results from our prime contractor's usage.

[AGREE] But it is only a start. It doesn't address technical issues and doesn't encourage improvement of a contractor's system engineering process.

[AGREE] However, it's hard to say at this point without it having more usage and without our understanding exactly what it is.

[DISAGREE] The model and the evaluation process still have problems. The most significant of

which is the lack of understanding about what the results mean. Changes made in the process model to get an extra "yes" don't mean much.

• **Round II—Group Opinion: Agree**

[AGREE STRONGLY] The clear connection between the improvement in software development and bottom line profits will be needed for contractors to make the required investment voluntarily. While contract awards are being based upon evaluation criteria that is not weighted on the contractor's proven performance and technical capability, this is a moot point.

2. Do you think a software acquisition organization within a SPO could use a tool such as the proposed SAMF to benchmark or measure their management process for self improvement?

• **Round I—Group Opinion: Yes**

[YES] Software Management needs lots of help here at Hanscom. However, without upper level management support for the tool, it won't help.

[YES] Yes, provided there is training on: 1. how to do the evaluation, 2. how to understand the results, 3. how to improve the process and not the results.

[YES] Pro—TQM metric and continuous improvement category make a black or white judgement call simple and uniform. Con—Standards, once accepted, are seldom used with discretion or the appropriate exceptions required for special cases.

[YES] It could objectively quantify capabilities and identify areas for improvement.

[YES] There is far too much ad hoc and seat-of-the-pants management of software at the SPO. This ruins the credibility of the folks working software issues since they are viewed as practicing a black art instead of managing a real entity. The problem compounds when they try to force the contractor to produce quality software — since they have no credibility it is impossible to force the contractor to step up to their standards. Some tool is essential to measure, report and make real progress in software management.

[YES] Both CMM and SAMF are designed to assess the presence of defined procedures. If correctly adapted to the acquisition management domain the

assessment of the process should be valid. It never hurts to take a look at one's organizational process. However, the danger with all assessments is that scoring well can often be more important than having a good process. The bottom line is that the acquisition itself is more important than all the planning and assessment.

[YES] I definitely think that an assessment of an organization's software acquisition management process helps to characterize its software acquisition management strengths and weaknesses.

[YES] But many SPOs are too lean to make time to do much besides put out fires.

[YES] Quite strongly agree. It also dovetails well with Dr. Deming's work. We have asked SEI about this and they said they working on some related topics.

• **Round II—Group Opinion: Yes**

[YES] However, the continued high rate of change in computer technology with the parallel evolution of "software methodology" [structured, object oriented, 4GL, etc.] provides a dynamic environment that may or may not lend itself to stochastic/process analysis.

3. Do you think the idea of adapting the SEI's CMM to the Air Force Software Acquisition Process is feasible?

• **Round I—Group Opinion: Yes**

[YES] The CMM is basically a philosophical approach to management. While the questions relate to software development, the fundamental philosophy is still the same. For example, in a newly established SPO, processes are not yet defined. As they are defined and understood, people must be trained on them. Further, the business environment changes as the system itself changes... e.g., moves to the next engineering phase, approaches turnover, etc. The philosophy and management approach must continually accommodate this.

[YES] I feel that capability evaluations will provide the Air Force with a method by which the rank of the overall organizations to produce software in a timely, repeatable fashion. However grading contractors will (may) create contractual implication that the Air Force will have to deal with.

[YES] It will depend on how the program is implemented. There has to be a training program. Just being able to evaluate the process isn't enough.

[YES] Many concepts in this area have been established in AFSCP 800-14 and 800-43 and the AFOTEC pamphlet series.

[YES] Pro—The military protocol system can adapt to any idea considered essential by senior officers/officials. Con—The reduction of defense manpower, number of specialties within the software engineering and acquisition management, the 15-17 years needed for mastery would be significantly increased if all necessary personnel receive the CMM training.

[YES] SEI's CMM seems ideally suited for this. There is already a methodology and data to support it. Face it, you have got to start somewhere or we will all just keep floundering flushing defense dollars down the toilette while we try to get our hands around this monster.

[YES] There are no other tools in place that I know of.

[YES] A formal process can be applied to any activity including acquisition and can, therefore, be specified, measured and controlled. A note of caution: The SEI's CMM can only assess the presence of documented processes; and, to a more limited extent, adherence to them. It doesn't assess the quality and correctness of those procedures. Generally the presence of procedures/processes leads to a higher probability of success when compared to the absence of documented procedures. However, the SEI's CMM and the SAMF must be carefully used as an indicator of trends in progress... they should not be used as an absolute rating scale.

[NO] The Key Process Areas from the CMM do not represent the important activities of the SPO. As to the 5 levels of maturity, I strongly disagree with their order. In my opinion, measurement (level 4) should be the first step in any improvement effort.

[NO] We (the Air Force) don't develop or modify or reuse software for "large" projects. The question areas of the SEI's CMM can be used/adapted for the Air Force software acquisition process—but, the questions don't apply directly.

[NO] You can use the basic concept of a capability model but the KPAs would be much different. The rotation system works against building two levels of maturity. The acquisition model is changing, as in

going to integrated product development. Software "engineers" in the SPOs need to have had experience as developers of large scale software systems, which they don't.

• Round II—Group Opinion: Yes

[YES] Although I do not change my answer, the comment about development experience is great. The Hardware people developed the Blue II program to enlighten engineers on maintenance issues. Software development and maintenance experience should be a personnel factor. Maybe we should cycle MAJCOM software maintainers through acquisition or IWSM?

[YES] a) Does the current knowledge base provide sufficient objective information to develop an Artificial Intelligence shell? b) Can training and new methods be so structured that Government imposition of yet another set of "standards" on our contractors is acknowledged as an initiative to bring the current "60's technology base" in software up to the level needed for world class products in 2000.

[NO] Perhaps if the SEI would change their CMM then everyone would respond, "YES."

4. To be able to benchmark or measure a process one has to be able to define the process. The Air Force Software Acquisition Process can be defined from current literature such as DoD directives and Air Force policy.

• Round I—Group Opinion: Disagree/ Agree

[AGREE] Can be defined only in the broadest sense.

[AGREE] It really depends on what you mean by "define the process." The CMM definition of the software development process is a very loose one. They identified only a few important activities.

[DISAGREE] DoD and Air Force guidance is available and fairly consistent however it's rarely followed consistently and doesn't reflect the "real world" software acquisition process.

[DISAGREE] Only to a certain level. Not to the level of truly enactable processes which require a lot of interpretation and additional work.

[DISAGREE] The Air Force software acquisition process is ad hoc at best. Every SPO does it differently (re-invent the wheel)... Really dumb!!

[DISAGREE] The process is in a state of flux what with IPD, CAID.

[DISAGREE STRONGLY] 2167A is a data delivery process. It's only usable after the contract is let. At least 50% of the important work happens before the contract is released. 2167A doesn't even address this period. IEEE has about the best material available.

[DISAGREE STRONGLY] Current lack of uniform controls on software acquisition processes is a balanced between the two models of military standardization and commercial practice. DoD directives and Air Force policy reflect a "minimalist" controls perspective appropriate to a "free market economy."

[DISAGREE STRONGLY] {This comment also applies to question 5.} This is true only at a top level. How a SPO is organized, how many products, how many customers etc. will all affect the processes. For example, there is no good reason to have the detailed process for all SPO CCBs to be the same; however, they must all have a CCB and a process. To try to define the CCB process for all organizations would be to raise the level of abstraction to the lowest common denominator. I doubt if meaningful measurements can be found on that. (Remember, meaning is real only at the low levels of management i.e., below the SPO director. AFMC standard process will not help the people who work with and within the process, but probably encourages meaningless comparisons between SPOs and Product Centers.) A higher level of abstraction will lead to more eddies of informal processes... if these aren't defined, controlled then the definition effort really hasn't made any gains.

• **Round II—Group Opinion: Disagree**

[DISAGREE] The definition of the elements of a standard software acquisition process clearly transcend the objectives of applying another unique metric.

[DISAGREE] DoD-STD-2167A only gives a structured approach on how to develop a product (software). This standard is also expected to be tailored to the bare minimal requirements necessary to meet SPO objectives (different for every project). The process of software acquisition is not specifically

stated anywhere. Normally you can glean bits and pieces of the "Acquisition Process" from other regs, and standards (AFR 800-14; MIL-STD-1521 et al), plus what experience the acquisition manager has had in the past (lessons learned) either Government or commercial.

[DISAGREE STRONGLY] Software Engineering expertise is not available within the SPOs even if the guidance was available. "Generic Engineers," "Banked Pilots," and "Acquisition smart O-6s" have no hope of managing an important/critical software acquisition. Well-trained software engineers with experience and up-to-date guidance are the only solution. Further, the guidance and training must address real acquisition problems/ issues not contractor monitoring issues. Guidance must address advanced planning.

5. There is a "standard" Air Force Software Acquisition Process used at all Air Force Material Command (AFMC) Product Centers.

• **Round I—Group Opinion: Disagree**

[DISAGREE] DoD and Air Force guidance is available and fairly consistent however it's rarely followed consistently and doesn't reflect the "real world" software acquisition process.

[DISAGREE] I don't think there is a standard even within a given Product Center.

[DISAGREE] There is limited evidence of the use of a standard; and, usually, differing applications have caused local interpretations to "color" the standard. Basically, the process must always be adapted to the program.

[DISAGREE] Not sure how different but have seen SOWs written differently by different commands. Have also heard some commands do not tailor standards and DIDS.

[DISAGREE STRONGLY] Most people at Hanscom think they know software acquisition. They lie.

[DISAGREE STRONGLY] Most product centers aren't consistent from program to program.

[DISAGREE STRONGLY] Software acquisition process is significantly different at ESD, SSD, BMO, and ASD depending on the influence of MITRE, Aerospace, TRW and Softec. The WR-

ALC F15 Radar Software is influenced by TRW while the WR-ALC Joint STARS software is controlled by MITRE.

• **Round II—Group Opinion: Disagree**

[DISAGREE STRONGLY] I agree very much with the comment about Mitre, TRW, etc. The worst part of this agreement is that none of the companies have any idea of real-life software development/maintenance. They know even less than most Government organizations think they do.

[DISAGREE STRONGLY] The risks associated with the product centers acquisitions vary. The importance of the software as an element in the "systems" acquisition determines the level of assurance technology applied [budget allocation] and subsequent breadth and depth of visibility available to the program office.

[DISAGREE STRONGLY] Within every product center the software acquisition process varies from SPO to SPO. Depending on the experience and training of the acquisition managers, the process implemented will run the gambit from laughable to well defined and disciplined.

6. The Air Force Software Acquisition Process is the same for all levels of programs (i.e. basket SPOs -vs- Major Program SPOs).

• **Round I—Group Opinion: Disagree**

[AGREE] At least at SSD until IPD became widespread.

[AGREE] However, basket SPO's require cross-functional expertise in (typically) one individual whereas major programs have the luxury of a few functional experts in each functional area.

[DISAGREE] My experience is that the smaller the project, the more likely it is that sound software engineering is done.

[DISAGREE] There is limited evidence of the use of a standard; and, usually, differing applications have caused local interpretations to "color" the standard. The process must always be adapted to different program applications.

[DISAGREE STRONGLY] Most product centers aren't consistent from program to program. Besides,

acquisition and engineering requirements may be very different.

[DISAGREE STRONGLY] No—external influences, environments, and many others affect what processes exist and what ones are critical.

[DISAGREE STRONGLY] Only major weapons systems and mission/safety of life or property can sustain the costs of a full military software acquisition; e.g. SQAP, IV&V, separate software CCB and material review boards, software managers, SDP/CPDP, CRISD, PDSS plans, etc.

• **Round II—Group Opinion: Disagree**

[DISAGREE] The current focus on providing a general functional specification to describe system performance does not provide the level of depth required to address the software existent at levels six and below in the MIL-STD-881B work breakdown structure.

[DISAGREE STRONGLY] Still strongly disagree. This statement also does not recognize the differences across system domains (i.e. Aviation, C3, etc).

7. The management of software acquisition can be defined as a stand alone functional discipline within a SPO.

• **Round I—Group Opinion: Disagree**

[AGREE STRONGLY] All category 1D programs should be manned with AFSC 2736 field grade officers from PMP approval through ALC depot activation and software responsibility transfer.

[AGREE] Important elements of it can—Remember, your goal is to improve the process, not to determine what the best possible process is.

[DISAGREE] Perhaps there should be an overall process manager who would coordinate processes that affect more than one part of the SPO, but many processes are (or should be) within one directorate or division of the SPO. If this is not the case, maybe the organization should be re-aligned.

[DISAGREE] Usually part of the system engineering shop.

[DISAGREE STRONGLY] It has to be cross-functional. Lots of people have lots of talents which must be "blended" together.

[DISAGREE STRONGLY] Needs to be linked to systems engineering, computer hardware technology, computer resources utilization measurements. [DISAGREE STRONGLY] Software acquisition must be an integral element of the number acquisition. Many of our past software difficulties can be tied to an inadequate understanding of the overall system's requirements and the allocation to software.

[DISAGREE STRONGLY] The software acquisition managers in a SPO, if they exist at all, are usually buried down in the bowels of the organization. It's very difficult to say that they are a "stand alone" discipline. Most PM's don't like the work software. It scares the hell out of them!!! I feel this is because none of them have the foggiest idea what software acquisition/software development is, and just how important software is to the proper/reliable operation of a system.

[DISAGREE STRONGLY] When functional areas don't talk/communicate, things don't work to-

gether. Besides software directly affects logistics (support planning), testing, data and configuration management, etc. Of these areas, logistics and software support planning are my number one concern.

• Round II—Group Opinion: Disagree

[AGREE STRONGLY] I would assert that the current crisis in software life cycle maintenance is the direct result of the lack of emphasis on software from the concept evaluation phase and failure of our current systems engineering and development process standards to apply adequate levels of control.

[DISAGREE STRONGLY] When our contract was awarded 14 months ago, there was one huge software organization. Now, software engineering has been "decentralized" to work cross-discipline concerns. Communication has really improved and people are working as a team. Non-software people aren't afraid of software anymore!

Comments on the Organization Level

SEI-91-TR-25 defines an organization as "a unit within a company, agency, or service that shares common management, is centered at a single geographical site, and has responsibility for a common business area." The scope of the software development organization may differ depending on the company size and assessment strategy (i.e. evaluating the maturity level of a company's division versus a unit within the division). Furthermore, the methodology for implementing the maturity assessment implies multiple projects should be evaluated to determine the organization's process maturity.

8. Considering the key points listed above, how should the CMM organization/project assessment focus be mapped to the Air Force software acquisition management process?

① **Target the SAMF process self assessment tool at the SPO level where the SPO is the**

organization and the segments/sub-systems/teams are the projects.

② **Target the SAMF process self assessment tool at the SPO level where the SPO is both the organization and the project.**

③ **Target the SAMF process self assessment tool at the AFMC Product Center level where the Product Center is the organization and the SPOs are the projects.**

④ **Target the SAMF process self assessment tool at the Program Executive Officer (PEO) level where each PEO program grouping is the organization and the SPOs are the projects.**

⑤ **Other recommended mapping.**

• Round I—Group Opinion: 3rd Choice

① Each SPO does business differently at SSD, as such each SPO at SSD does software development differently. The PM of all the SPOs at SSD sets the "tone" as to what the development strategy will be (will software be an important part or will it be relegated as "not very important"). I feel it should start at the SPO PM. If the PM feels software is important then the job may be done correctly. If the

PM feels software is not important then very little attention will be paid to software acquisition.

① The target level must be aimed at the SPO. Otherwise, no one will pay any attention to the findings of evaluating the process.

② Cannot go any higher then SPO level because the SPOs are too diverse to manage as an organization.

③ The differentiation between the first two is probably more dependent upon the SPO than anything else. I do not believe this should be limited to software. Fundamentally, the concepts for success are the same. Also, many times a customer's requirement can be met by either hardware or software (or procedural). How are those trades made? Both processes required discipline.

④ You will run into problems with the basket SPOs. By emphasizing the product, the process can be monitored more closely.

⑤ SAMF should be adapted to product types by taking into account broad product characteristics (which is why product centers exist in general.) Then based upon the model of broad product characteristics there should be further refinement/tailoring for specific program needs/peculiarities.

⑥ Seems like the logical level — if you go to the SPO level you will get too much disconnect while everyone is looking for some standardized methodology. At the product center you can learn from all SPOs while leaving them some amount of autonomy to customize as they see fit.

⑦ The Product Centers generally share a common application domain, and therefore, should have a reasonably consistent acquisition process. The SPO as a project provides an individual instance of the

acquisition process to be applied. This provides sufficient samples to determine the acquisition process maturity. However, just as each software development organization will adapt the CMM to its organization structure, so should each acquisition program. The size, internal structure, and domain of programs vary too much for a single approach to be successful.

⑧ To make an impact the SAMF needs to go beyond what are at times short term objectives of a SPO.

⑨ Target the SAMF process self assessment tool at Air Combat and Air Mobility command users where AFMC Product Divisions are the Project. The rationale for the above is that the basis of all operational needs is the using command. Unless the 49XXs company/field grades require the product divisions 2736s to be responsive to the mission needs, there can be no sustained support for the use of the CMM Software Acquisition process assessments.

• Round II—Group Opinion: 3rd Choice

① I suspect the people who responded 4 or 5 believe the world is flat.

② Product centers can't keep track of the total program, let alone its software. Additionally, many centers support and acquire multiple domain areas.

③ The implementation of the clear accountability in design process as a sub-element of CASE may prove to be a solution to locating the CMM focus, if implemented between the using and implementation command prior to systems acquisition. The difference in the product centers application domains will make this hard.

Comments on the Definition of the Acquisition Environment

Within the stated context of use several key bounds and constraints have not been addressed:

- The acquisition of a product does not apply specialized engineering knowledge domains in the same manner that the design, development, manufacturing development, production, deployment and post deployment support do.
- The targeted "organization" and "project" being one and the same may violate the basic rules of

spatial and temporal separation in both "organization theory" and "management span of control."

- Can the premise for separation of software acquisition management be substantiated in light of the historical and technical problems of timing and sizing?
- Can overall PEO/DAC management authority and the use of systems engineering trades for

system optimization allow MCCR separate management responsibility/authority when cost and schedule is the basis for acquisition program performance?

- Reference the third bullet—With IWSM I'm not sure this is a good assumption. This model should support all phases.

Comments on the Maturity Levels

Level 1—Initial: *The software process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort.*

9. Is this maturity level applicable to the proposed SAMF?

- Round I—Group Opinion: Yes

[YES] For example, how are budgets defined? How are schedules adjusted? How does a CCB function? All these need to be defined.

- Round II—Group Opinion: Yes

10. The Air Force Software Acquisition Process is based on existing directives and regulations (i.e. DoD 5000.1 & DoD 5000.2). However, even when these "minimum" management processes are implemented, the software acquisition process can still be ad hoc.

- Round I—Group Opinion: Agree

[AGREE STRONGLY] Because standards and DIDs can be tailored to be meaningless. Also, software parts of contracts can contain gross errors/oversights.

[AGREE STRONGLY] None of the existing regulations provide much/any real guidance. Most Air Force programs are chaotic.

[AGREE STRONGLY] There is no way that high level directives can provide sufficient detail for every program.

[AGREE] Because it seems every directive or regulation has an "out" for those clever enough to find it.

- Round II—Group Opinion: Agree

[AGREE] Contract negotiations after contract award can undo the "good/minimum" management practices contained in these regulations.

Level 2—Repeatable: *Basic project management processes are established to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.*

11. Is this maturity level applicable to the proposed SAMF?

- Round I—Group Opinion: Yes

[YES] However, most SPOs seem to disband after projects and don't move on to a new major program.

[YES] Repeatability is "essential" to post deployment software support, reuse of development and test tools, software maintenance's "design recovery" or "reverse engineering" phase.

[YES] That is, given changes such as extending tours, creating applicable processes, and creating ownership, which includes accountability, are in place.

- Round II—Group Opinion: Yes

[NO] The focus on repeatability is justified if, and only if, software project successes are to be achieved in the future as they were in the past. Past failures must lead to cause future failures if the process is repeatable.

12. This maturity level implies a "repeatable software acquisition project management" capability. However, there is nominally only one "project" within a SPO, hence, the capability to

repeat the procurement of a similar project is not an important attribute of the SPO.

• **Round I—Group Opinion: Disagree**

[AGREE] But, there are modifications, ECPs, and multiple CSCI's to manage.

[DISAGREE] Most projects acquire mini-projects via ECPs or block upgrades.

[DISAGREE] Not necessarily true for launch vehicles.

[DISAGREE] Should use the same process or subset for all projects regardless of SPO, why re-invent the wheel?

[DISAGREE] The repeatability aspect has to do with the ability to formally define and apply a process, not necessarily the same exact one. Many programs have multi-phased procurements that require a new process for each phase.

[DISAGREE STRONGLY] Most SPOs have many "projects," all part of the main project.

[DISAGREE STRONGLY] We must learn from our successes and experience and teach/educate others in acquisition.

[DISAGREE STRONGLY] While there may be one project, individual parts of the project (e.g. contract changes, FCA/PCAs, CCB actions) are repeated many times.

• **Round II—Group Opinion: Disagree**

[DISAGREE STRONGLY] The concept/design/development/implementation process provides repeated opportunities to assist the contractor to focus/refocus on the characteristics and the attributes of the post deployment software support environment. The continued involvement of the contractor and user/customer should always be managed as a component of a repeatable software acquisition process.

13. If this maturity level is not applicable, the KPAs from this level should be integrated in another maturity level.

• **Round I—Group Opinion: Agree**

[AGREE] But remember what Watts said about skipping levels (you can't). Don't see how metrics could be Not Applicable!

[DISAGREE] I disagree with your approach. You should define your model top down, decide what are your maturity levels and what they mean, then define corresponding KPAs.

[DISAGREE] Because SPOs do not develop software nor maintain software.

• **Round II—Group Opinion: Agree**

[DISAGREE] The elements of the "repeatable software acquisition project management" process stem from rigorous application of good engineering practices not a separate, uniquely identifiable MCCR activity. The application of scientific process demands repeatable results—we haven't got there yet.

Level 3—Defined: The software process for both management and engineering activities is documented, standardized, and integrated into an organization-wide software process. All projects use a documented and approved version of the organization's process for developing and maintaining software.

14. Is this maturity level applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] Even though the process may be tailored for each SPO the same basic process is used.

[NO] Because SPOs do not develop software nor maintain software.

[NO] Until the "universal computer program" or "single algorithm solution" for all problems can be discovered a single process for developing and maintaining software remains more fantasy than fact. Please "consider" the differences between the "military standardization" model and the three varieties of "best commercial practice" that exist.

• **Round II—Group Opinion: Yes**

[YES] Got stuck here because SEI question deals with software process whereas SAMF is to deal with software acquisition process. It would have helped me to have your questions in terms of SAMF. But, the really big problem is there already are policies

and regulations that do much the same and SPOs are "in compliance"—that is, the IG would be satisfied that they were being followed, when they weren't. What would change if you re-documented them?

[NO] Until the PEO or Product Center Commanders are constrained to document policies and procedures, no set of rules can be standardized and integrated into the process. The national "Range Commanders Council" have documented and established a set of criteria, agreed to by all interactive "users," that have resulted in numerous universal protocols, e.g. IRIG, UTC, etc.

15. The characteristics of tracking cost, schedule, and functionality originally in the repeatable level should be integrated into the defined level.

• Round I—Group Opinion: Agree

[AGREE] It's a building block.

[DISAGREE] There is a degree of "defined process" at both SEI levels 2 and 3. At Level 3, defined process is comprehensive. Same applies to acquisition.

[DISAGREE STRONGLY] Cost and schedule reporting exist in "industry management" but have not shown significant results.

[NO OPINION] Defined implies repeatable, but it is still more advanced than just repeatable. There must also be more sophisticated/mature means of changing the repeatable process.

• Round II—Group Opinion: Agree

[AGREE] Software metrics can really help accomplish this.

[DISAGREE STRONGLY] The strength of the "open market" derives from competing process and methods producing a crucible that purifies and produces a single best product to capture market share from competitors; who must in their turn improve to survive. National defense is an extension of the political process that does not provide for change as rapidly as the commercial market. (Assembly language is not maintained the same way that Icons are.)

Level 4—Managed: Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled using detailed measures.

16. Is this maturity level applicable to the proposed SAMF?

• Round I—Group Opinion: Yes

[YES] Needs to be broader than just the "Software Process." It must apply to the SPO as a whole.

[NO] Because SPOs do not develop software nor maintain software.

• Round II—Group Opinion: Yes

[YES] I need the question reworded—what SPO "product" would you measure the quality of? The SEI is talking about the software product. You're not going to measure software quality again?

[NO] No objective proof—the juice is worth the squeeze. Standard conservative complaint of the "out-of-date."

17. This maturity level implies qualitative data is gathered on the Software Acquisition Process.

**• Round I—Group Opinion: Agree/
Agree
Strongly**

[AGREE STRONGLY] But what measures do they use and where do they document them? It's hard enough to find guidance on contractor-oriented metrics.

[AGREE STRONGLY] This is a very important step at which you are attempting to validate that your "defined" process is indeed a good one. Keep in mind that you might have a well "defined" bad process.

[DISAGREE] Process quality "control" can be derived from several reliability algorithms—unfortunately software reliability remains "undefined" or "standardized," as does the term "software maturity."

• Round II—Group Opinion: Agree

[DISAGREE] I think it is quantitative data on quality that is gathered.

[DISAGREE STRONGLY] The difference in management style of the SPO leaders, the level of MCCR knowledge in the SPO and contractor as well as what the basis of evaluation should be (i.e. software maturity at IOC, SLOC change percent per month during PDSS, design-to-cost, etc.) makes this a topic for further research.

Level 5—Optimizing: Continuous process improvement is enabled by quantitative feedback from the process and from testing innovative ideas and technologies.

18. Is this maturity level applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] Given unlimited resources and national priority for research and development this could be sustained for 10% of the SPOs (maybe); but the "value added" would make it hard to prove to cost accountants.

[YES] But like the other levels, it requires a matching commitment or contractual obligation with the developing contractor. If they don't collect the data it's tough for us to.

[NO] The Air Force experience I have suggests that this level is not attainable due to the way the Air Force is structured and operates.

• **Round II—Group Opinion: Yes**

[YES] Controlled assignments, advanced education and supportability versus cost and schedule are points to begin.

[NO] Not as stated, at least I can't think of any quantitative feedback. I would be concerned about selling up an "apparatus" that appears to collect meaningful data.

19. This level implies quantitative data is taken on the process. Gathering quantitative data takes time, how-

ever, with the new Integrated Weapons Systems Management acquisition process, opportunities for SPOs to take quantitative data will exist.

• **Round I—Group Opinion: Agree**

[AGREE STRONGLY] If full control of MCCR is provided, then recording would be needed on the 30-80% of the program life cycle costs expended in this area.

[AGREE STRONGLY] We do it now.

[AGREE STRONGLY] We do it! We are skeptical of proposed changes that are not supported by data or we may make the change if the portion of the process is one that lacked definition. (Hence it really isn't a change to a process, but a definition of it.) But if a change is proposed, we look for the data that tells why it should be made and what will indicate success or failure of the change.

[AGREE] But what measures do they use and where do they document them? It's hard enough to find guidance on contractor-oriented metrics.

[DISAGREE] Shouldn't the intention be to focus more on responding to new approaches and techniques and technologies as part of process development? This implies that you must have a process for managing "evolutionary" processes.

[NO OPINION] Don't know enough about IWSM, but I think my answer to 18—But like the other levels, it requires a matching commitment or contractual obligation with the developing contractor. If they don't collect the data it's tough for us to.—is the real key.

• **Round II—Group Opinion: Agree**

[AGREE] There are a lot of "we do it," but who gets the data? Data no one sees is wasted.

[AGREE] Full funding for the Joint Logistics Chiefs center at RADC can provide this data for every program from an "acquisition tax" at OSD level.

[DISAGREE] Not as stated, at least I can't think of any quantitative feedback. I would be concerned about selling up an "apparatus" that appears to collect meaningful data.

Comments to the CMM Key Process Area

Requirements Management involves establishing and maintaining an understanding and agreement with the customer on the requirements for the software throughout the software life cycle. The agreements cover both the technical requirements for the software and the nontechnical requirements, such as delivery dates for the software. The agreements form the basis for estimating, planning, performing, and tracking the project's software activities.

20. Is this Key Process Area applicable to the proposed SAMF?

• Round I—Group Opinion: Yes

[YES] This is the most important KPA in the Software Acquisition Process. I would divide it into several KPAs in fact.

[YES] This essentially defines the interaction between the acquisition organization and the developer.

[YES] There absolutely must be a defined approach for identifying requirements and changes to them and how the commitments will be met!

[NO] SPO doesn't generate software requirements.

• Round II—Group Opinion: Yes

[YES] With the use of rapid display prototyping, high level mathematical modeling and a user/implement high level concept of operations "what if" inputs to an Artificial Intelligence development CASE tool, the requirements area would become controlled.

[NO] Re: Software Technical (Functional) Requirements. The software requirements are derived from the system requirements by the developer not the SPO (Acquisition Agency). As such, the SPO must manage at the systems level. As the system level requirements change so will the flow-down to the software requirements level. The SPO needs to assume that the software requirements are testable, understandable and complete; it has no business in defining software requirements. Since the system level requirements change so often, the software requirements will be very unstable at best, through-

out the life cycle. The SPO must be kept up-to-date by the developer as to all new/modified software requirements.

21. Requirements Management involves establishing a working relation with the user (customer). This includes understanding and planning for the implementation of user requirements.

• Round I—Group Opinion: Agree Strongly

[DISAGREE STRONGLY] SPO doesn't generate software requirements.

• Round II—Group Opinion: Agree Strongly

[AGREE STRONGLY] Remember that the maintainer/supporter of the system is also a SPO customer.

[AGREE] However, it can be very difficult getting the user to articulate their requirements!

22. Requirements Management involves establishing a working relation with the contractor (seller). This includes contracting for and maintaining correct allocation of user requirements.

• Round I—Group Opinion: Agree Strongly

[AGREE STRONGLY] But where do the requirements people come from? The only course available in this area is a two week class at AFIT. Most SPO's have no software experts, let alone requirements experts.

[AGREE STRONGLY] ..."Contracting for and maintaining correct allocation of user requirements" ...can imply both requirements "creep" and developing "contractor advocacy." My opinion and recommendation would focus and clarify the software acquisition management model as based on fair

commercial practices and government case law, not military/industrial complex cronyism.

"...maintaining correct allocation..."

"...establishing a working relationship with the contractor (seller)." Semantics swing the response from disagree to agree, not the logic of the postulate.

[AGREE STRONGLY] {Comment also applies to questions 21 and 22} Why not form one description for requirements management here? What about things like testing for requirements internal consistency and testability?

[AGREE] But mostly it's defined by the contractual relationship that defines the initial requirements and the design and implementation process that results in a final product. The contract must allow for a process that acknowledges requirements that may change (ECPs) and they (the contractors) must create a design that allows change.

[AGREE] But SPO doesn't generate software requirements. Here you seem to be using a CMM KPA in a different fashion. The practise would have to be different than those in CMM document.

- **Round II—Group Opinion: Agree Strongly**

[AGREE STRONGLY] Requirements management during design must involve the maintainer as the interface spokes-person first and user second—the money spent on maintenance robs the purse that funds new developments and technology.

Software Project Planning involves developing estimates for the work to be performed, establishing the necessary commitments, and defining the plan to perform the work. A plan is established to address the commitments to the customer according to the resources, constraints, and capabilities of the project. The plan provides the basis for initiating the software effort and managing the progress of the work.

23. Is this Key Process Areas applicable to the proposed SAMF?

- **Round I—Group Opinion: Yes**

[YES] But can use some tailoring.

[YES] CPDP/SDP/SEMP

[NO] SPO doesn't have resources to do this. For example, to do the estimating could require prototyping part of the system.

- **Round II—Group Opinion: Yes**

[YES] Failure to begin software acquisition project planning at the XO/XR-Advance projects level unfairly limits the CONOPS and Post Deployment software support orientation that is essential at the beginning of the project/program.

24. Software Project Planning involves ensuring the contractor's software development plan is adequate and feasible.

- **Round I—Group Opinion: Agree Strongly**

[AGREE STRONGLY] Except the SEI's understanding of what is an "SDP" is not the same as the DI-MCCR-80030A.

[AGREE] That's part of it.

[AGREE] But this is too vague. Must insure that development process is well defined, reflects good use of state-of-the-art tools and techniques, provides for adequate QA, PA, TE, etc.

- **Round II—Group Opinion: Agree Strongly**

[DISAGREE] The SDP data item is deliverable as part of the CDRL; e.g. after contract award. This factor should be a primary technical evaluation element of source selection. Limits on bid and proposal funding must clearly delineate the dollar amount for the SDP preparation based on technology and complexity of the software.

25. Software Project Planning involves addressing the government activities necessary to implement the contractor software development plan (e.g. a detailed plan explaining the government technical review process of deliverables).

- **Round I—Group Opinion: Agree Strongly**

[AGREE STRONGLY] Anecdotal evidence on the need for in-depth government software project planning is substantiated by the number of large software projects that come in below cost and early. I would assert that unless the government software project planning begins with a program work breakdown structure, produces both a SEMP and TEMP before the preliminary contract work breakdown structure, as well as defining clear requirements/test pass-fail criteria, and source selection factors, standards and evaluation criteria, that "planning" is only for checking off the block.

[AGREE STRONGLY] But sounds likely you've switched tracks.

[AGREE STRONGLY] But, this is not always done.

[AGREE STRONGLY] There is no such plan today.

[AGREE STRONGLY] This could be part of the system engineering management plan.

[AGREE] Also, what will the feedback loops be? How will the developer be measured exactly?

[AGREE] And there should be a software risk management plan (SRMP).

• **Round II—Group Opinion:**
Agree/Agree Strongly

[DISAGREE STRONGLY] The explanation of the government document technical review process is not quite as unnecessary as Tits on a Boar Hog, but darn close. What is needed is both a SPO and contractor orientation toward a highly reliable and maintainable product that can support military operations (Ao) essential to national defense. Data item problems stem from three easily identifiable sources; e.g. 1) lack of quality assurance requirements, 2) Government technical direction of contractors efforts in too great a depth, without ensuring adequate time and timing associated with mile/inch stones, and 3) amateurism in the Government program/project management staffing of sufficient resources when needed.

Software Project Tracking and Oversight involves tracking and reviewing the software accomplishments and results against documented estimates, commitments, and plans, and adjusting these based on the actual

accomplishments and results. A documented plan for the software effort is used as the basis for tracking the software activities, communicating status, and revising plans. The software activities are monitored by the software managers on a regular basis. Regular technical reviews and reviews with the project manager and senior management are conducted to ensure that management and staff are aware of the software status and plans, and that issues receive appropriate attention.

26. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] And should be part of the SRMP.

[YES] One must keep in mind the axiom "that which isn't tracked will not be completed"—if it is not important enough to track, the signal management sends is that the schedule commitment for that element isn't important. Further, how will one obtain the data to support changes to a process?

[NO] Why duplicate contractor's efforts? Also, SPO doesn't have resources to do this.

• **Round II—Group Opinion: Yes**

[YES] The developer keeps track of his development effort by referring to the SDP and tracking accomplishments to this plan. I feel the SPO should also use the contractor SDP as a tool to assure the contractor is following all the commitments made in the SDP (Government approved document).

[NO] With full control of and responsibility for MCCR, as given, software project tracking and oversight become the basis for process improvement (TQM) and training requirements metrics.

27. Software Project Tracking and Oversight involves tracking and reviewing the accomplishments of the contractor in accordance with the contract.

• **Round I—Group Opinion: Agree**

[AGREE STRONGLY] Also the SDP and the SEMP.

[AGREE STRONGLY] Also, subcontractors!

[AGREE STRONGLY] Hopefully, the contractor tracks this and the SPO reviews, understands, and helps overcome obstacles.

[AGREE] But this is vague.

[DISAGREE STRONGLY] Involves the system for tracking the contractor and the level to which the system is applied.

[DISAGREE STRONGLY] The application of the SAMF must work if it is to be successful—therefore, the tracking and oversight process cannot be based on a set of “accomplishments of the contractor in accordance with the contract” as currently done. **Rationale:** Contracts SOW and CDRL are written to explain “what the end product/material/service is ‘not’ how it is to be designed/developed/performed.” Effective evaluation is based upon quantified performance criteria to determine “how well” the contractor is, in fact, designing/developing/performing. The dichotomy exists in that the contract exist to specify the “what” and the tracking and oversight KPA must objectively evaluate the “how.”

• **Round II—Group Opinion: Agree/
Agree
Strongly**

[AGREE STRONGLY] Tracking must emphasize more than just delivery of data items.

[DISAGREE STRONGLY] An “instant contract” includes working agreements as well as interpretations of implicit requirements and derived requirements—the evaluation of accomplishments must apply mature judgement to determine equitable satisfaction of the contract.

28. Software Project Tracking and Oversight involves tracking and assessing the software acquisition management process in accordance to the Software Project Plan.

• **Round I—Group Opinion: Agree**

[AGREE STRONGLY] Also the SDP and the SEMP.

[AGREE] This is also vague.

[DISAGREE STRONGLY] Any “plan” is only a “snap-shot in time” of goals and objectives current when the plan was drawn; tracking and assessing

need to address whether the contractor’s “how” can provide the “what.”

• **Round II—Group Opinion: Agree**

[DISAGREE] This is taking the “wet-noodle” approach. A more effective tool is a contractually binding Integrated Master Plan and Integrated Master Schedule, which includes major events and accomplishment criteria. This is what the contractor proposes during source selection, and subsequently must follow.

[DISAGREE STRONGLY] The “tracking and assessing” need to be targeted at the identification and resolution of the reliable, supportable weapon system MCCR development process problems. Additional, uncommitted time and resources can be allocated to an ordered list of “druthers,” maintained by SPO software cadre.

Software Subcontract Management involves selecting a software subcontractor, establishing commitments with the subcontractor on the work to be performed, coordinating activities with the subcontractor, and tracking and reviewing the subcontractor’s performance and results. The subcontractor is selected based on its ability to perform the work. A documented agreement covering the technical and nontechnical (e.g., legal, financial, and administrative) requirements is established and is the basis for managing the subcontract. Regular technical and management reviews are conducted to ensure that management and staff of both organizations are aware of the software status and plans, and that issues receive appropriate attention.

29. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] Although privity of contract prevents direct interaction with subs; and you must depend on subcontract flow-down.

[YES] For your purposes, the “subcontractor” management should be raised to overall “contract management.” The expectations and overall management requirements are generally the same for SPO→Prime as for Prime→Subcontractor.

[NO] Because this is a contractor's task.

[NO] Government is limited to subcontractor management. Subcontractor "monitoring" might be more appropriate.

• **Round II—Group Opinion: Yes**

[YES] Subcontractor Quality Assurance is available as a standard element of the SOW (MIL-STD-1535B?). This mechanism responds well when teamed with costs of software quality programs under MIL-STD—for scrap and rework/non-conforming items/material control/review boards.

[NO] The contractor is ultimately responsible for sub-contracted work, and should be treated as such. We should not have to get into their knickers!

30. Subcontract management is a function of the prime contractor. This should not be management directly by the SPO. Instead, subcontractor management requirements should be stated clearly in the prime contract.

• **Round I—Group Opinion: Agree**

[AGREE STRONGLY] However, government participation in prime contractor's subcontractor reviews is a real must to give the government insight as to "what is really going on."

[AGREE STRONGLY] But you still need a monitoring process.

[DISAGREE] You should state obligations the prime should fulfill and he is then charged with flow-down and subcontract management schemes that result in the fulfillment of the prime contract.

[DISAGREE] Although Subcontract Management must not be controlled by the SPO, it should be closely monitored.

[DISAGREE] The management of important and sometimes critical subcontractor elements of the system being acquired must be considered as a KPA. The SPO should have the ability to interface with these KPA subcontractors along with the Prime. Direct access is important for the sake of obtaining timely and accurate software status information. (Especially where sub is large scale developer, which often happens.)

• **Round II—Group Opinion: Agree**

[DISAGREE] This question is slightly contradictory in that if a subcontract management requirements is stated in the prime contract then the SPO, who manages the prime, is managing the subs. The privity of contract principle says the SPO does not interact directly with the subs, but uses the prime as an agent. What this means then is that we state our expectations for the entire product and the prime must flow-down those expectations or it won't fulfill the requirements of the prime contract.

[DISAGREE STRONGLY] Software developed by each contractor is different—its integration and satisfactory delivery is the prime contractor's responsibility. The ensurance of the process is the implementor/SPO responsibility; ask any user/customer!

Software Quality Assurance involves reviewing and auditing the software products and activities to ensure that they comply with the applicable processes, standards, and procedures, and providing the staff and managers with the results of their reviews and audits. The software quality assurance function is required on all projects. The group performing this function is independent of the software groups and project management. A senior manager who is committed to handling all major software quality assurance issues is identified. Where compliance issues exist, the software quality assurance group works with the appropriate managers, including senior management where required, to resolve the issues.

31. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] To IV&V or not IV&V. That is usually a lost cause.

[YES] A contractor function.

[YES] There should be a quality assurance function within the SPO that reviews processes and products. Though these are not software, they are program management items that are needed for success.

• **Round II—Group Opinion: Yes**

[NO] Industrial engineering control systems such as Software Quality Assurance are best when applied after product specification authentication. Software engineering control systems such as Independent Verification and Validation or Red/Tiger Teams seem to work better in the less structured early engineering phases.

[NO] Quality Assurance is an activity that should be an integral part of contract management, and not a separate task. This is very un-"TQM"-ism. I don't think SQA should be a separate KPA even in the CMM.

32. Software Quality Assurance is a function of the contractor's software development organization and should not be managed by the SPO as a separate distinct discipline.

• Round I—Group Opinion: Agree

[AGREE] SQA should not be "managed" by the SPO but it should be monitored and evaluated by SPO for effectiveness. Monitoring of the effort may be implemented through a variety of agencies (IV&V, DCAS, etc.).

[DISAGREE] 1. No SQA should be independent of engineering. 2. SQA is integral to software management, not separate.

[DISAGREE STRONGLY] IV&V is a must!

• Round II—Group Opinion: Agree

[DISAGREE] Still disagree. While SQA should be part of the software process, it should be independent (like SCM) from the engineering/development group.

[DISAGREE STRONGLY] The software QA reviews of data items prior to contractor release is a failure; it's mirrored by the inspection of incoming data items failure by the SPO's SQA.

[DISAGREE STRONGLY] Tough question—my answer means the SPO should manage its internal QA separately.

33. It is the function of the SPO's Software Project Management to ensure that the SQA function of the contractor is properly performed.

• Round I—Group Opinion: Agree

[AGREE STRONGLY] This is only one function of the SPO's SPM. Also, to make sure it is adequate in scope and introduces no conflicts of interest in developer organization.

[DISAGREE] DPRO's responsibility.

• Round II—Group Opinion: Agree

[AGREE STRONGLY] I need an SAMF/CMM translator for this one.

[AGREE] Interesting comment on the DPRO. Will DPROs be measured under this system? They aren't part of the SPO or product center so who measures their capability? My further concern is that most DPROs have less software training/experience than SPOs do.

[DISAGREE] I suppose most respondents don't deal with DPROs. I suspect that is why more people did not disagree with this item. (I still think it's the DPRO responsibility.)

Software Configuration Management involves selecting project baseline items (e.g., the project description, products, and process specifications of the project), controlling these items and changes to them, and recording and reporting status and change activity for these items. Changes to these baseline items are controlled systematically using a defined change control process. The configuration (software and documentation) of a system, or of any of the controlled intermediate or support products, can be distinctly identified at any point in time.

34. Is this Key Process Areas applicable to the proposed SAMF?

• Round I—Group Opinion: Yes

[YES] For the government controlled baselines.

• Round II—Group Opinion: Yes

35. Software Configuration Management is a function of the contractor's software development organization and should not be managed by the SPO as a separate discipline.

• **Round I—Group Opinion: Disagree**

[AGREE] Not managed, but monitored.

[DISAGREE] 1. CM should not be part of engineering. 2. Agree for configuration identification, not for change control and data.

[DISAGREE] By definition true. CM involves the SPO or at least the product organization. This is not the same as developmental configuration control.

[DISAGREE STRONGLY] Anecdotal evidence on the need for in-depth government software project planning is substantiated by the number of large software projects that come in below cost and early. I would assert that unless the government software project planning begins with a program work breakdown structure, produces both a SEMP and TEMP before the preliminary contract work breakdown structure, as well as defining clear requirements/test pass-fail criteria, and source selection factors, standards and evaluation criteria, that "planning" is only for checking off the block.

• **Round II—Group Opinion: Disagree**

[AGREE] This is the second question that I have gotten hung up on semantics. Management versus Control versus Monitoring. All three are loaded terms which mean different things to each reviewer. I do believe the SPO has a strong CM role/interest. Defining that role is the issue.

[DISAGREE] I changed my answer. I must have overlooked "contractor's" the last time I read the question.

36. It is the function of the SPO Configuration Management Directorate, in close coordination with the project and engineering directorates, to ensure the SCM function of the contractor is properly performed.

• **Round I—Group Opinion: Agree**

[AGREE STRONGLY] But also must include planning for transition to government SCM at acceptance.

[AGREE STRONGLY] The SPO CM office also keeps track of all product deliverables (documentation software versions etc.).

[AGREE] Again, DPRO is involved.

[AGREE] But it's more important that there is active participation on reviewing and approving changes to configuration items.

[DISAGREE STRONGLY] The management of requirements allocation from the functional specification by-and-large sets the basis of design and the set/sub-set(s) of software requirements aggregated for configuration items. This activity determines the basic components of the "product" and "sets" the post deployment software support costs and schedules for the product "life cycle." The "configuration management" elements of this KPA should be performed by the user versus the SPO or contractor.

• **Round II—Group Opinion: Agree**

[AGREE] Need translator.

Organization Process Focus involves developing and maintaining an understanding of the organization's software processes and coordinating the activities to specify and improve these processes. A group such as a software engineering process group acts as the focus for the software process activities in the organization. This group coordinates the projects' software process definition activities and the organization's long-term process improvement efforts.

37. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] But probably a lesser function.

[NO] This KPA should be thrown out.

• **Round II—Group Opinion: Yes**

[YES] But you need to address the rotation of folks in and out of this activity wherein the group resides—this KPA takes legacy/lessons learned.

[NO] The right way to evaluate an organization's process development is to have one "process" KPA at each level. Level 3 currently has 2 "process" KPAs. This particular one does not provide any additional information to what's obtained through the "Process Definition" KPA. I think this KPA should be thrown out of both the SAMF and the CMM.

38. Organizational Process Focus implies a software acquisition process group acts as the focal point for process improvement. However, implementation of such a group at the SPO level is not feasible.

- **Round I—Group Opinion: Disagree**

[AGREE] Are you sure it's not feasible at the SPO level?

[AGREE] Potentially more manpower needed.

[DISAGREE] Depends on what or who's processes that you're trying to manage.

[DISAGREE] SPOs must be responsible for self improvement, but product center focus, Air Force focus, and DoD focus are also needed. Lessons learned.

[DISAGREE STRONGLY] We have such groups and they actually work (from time to time)!

[DISAGREE STRONGLY] With control of all MCCR resources it is feasible.

- **Round II—Group Opinion: Disagree**

[AGREE STRONGLY] Tough question—my answer means the SPO should manage its internal QA separately.

39. A Software Acquisition Process Group (SAPG) would best be implemented at the AFMC Product Center level.

- **Round I—Group Opinion: Agree**

[AGREE] As a general guidance/EN function however each SPO must tailor to suit their project.

[AGREE] Don't know about phrases like "best" or "only," but process groups can be implemented at several levels.

[DISAGREE] Implement at all levels or DoD will end up with a million bad ways to do the job.

[DISAGREE STRONGLY] 1) Should not be limited to software—must be management of total project. 2) Must be a program manager level—he runs the process, has authority to make changes, and is accountable for the results!

[DISAGREE STRONGLY] Should be an SAPG at the product division level with representatives from each SPO.

[DISAGREE STRONGLY] The SAPG would best be at the Air Combat/Air Mobility Commands. Consider embedding product centers in the operating commands as was the practice when we had an AFMC before.

- **Round II—Group Opinion: Disagree/ Agree**

[AGREE STRONGLY] But you would have to staff the group—is the pay-off really there?

Organization Process Definition involves establishing and maintaining a standard software process for the organization, along with related items, for use by the projects in establishing their software process. This standard software process provides a common process for software development and software maintenance projects. It defines the essential process steps for the projects and establishes the common basis for measuring process performance across all projects and for long-term improvement.

40. Is this Key Process Areas applicable to the proposed SAMF?

- **Round I—Group Opinion: Yes**

[YES] Within domain areas, example: OFPs, command and control, etc.

[YES] Yes, but to lesser function.

[NO] This KPA should be combined with the previous one (Organizational Process Focus) even in the CMM.

[NO] The very nature of developing software is very volatile. There is no one methodology or standard software process.

- **Round II—Group Opinion: Yes**

41. Organizational Process Definition implies all product center SPOs implement a standard software acquisition process.

- **Round I—Group Opinion: Agree**

[AGREE] Probably agree since, as stated earlier, each product center should basically share a common application domain; however, it's tough to believe there is a "standard software acquisition process." Each program must be handled differently, it would seem.

[DISAGREE] Approaches must be tailored from standard guidelines.

[DISAGREE] Every program is different. Thus, a standard process would turn into "square filling" and would lose the benefit of being a process that gets better over time.

[DISAGREE] Only true if the product center works within one domain (i.e. OFPs, command and control, etc.).

[DISAGREE STRONGLY] Can be usable if and only if mature decision making can tailor/waive part or whole process.

[DISAGREE STRONGLY] It is customer driven! Not all SPOs have the same customers. Give the authority and responsibility to program managers!

• **Round II—Group Opinion: Disagree/ Agree**

[AGREE STRONGLY] This would be best given the rotation of personnel from one product center to another. It would also be best for the contractors.

[DISAGREE] The application domain is the driving factor for development and support planning. BMO buys one type of weapon system. ESC, SMC, and ASC cross domains and will have problems with one approach.

Training Program involves identifying the training needs of the organization, the projects, and the individuals and developing and procuring training courses to address these needs. The training program ensures that training needed to perform each of the organization's job functions is appropriate and is not circumvented inappropriately.

42. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] This item is the key to the Air Force's problem.

[YES] Needs to cover process within the SPO: technical training on system being acquired/supported; APDP Acquisition training; and, familiarization with customer.

• **Round II—Group Opinion: Yes**

43. A SPO Training Program should address the organization's general acquisition training requirements (e.g. training needs to meet the Acquisition Professional Development Program requirements).

• **Round I—Group Opinion: Agree/ Agree Strongly**

[AGREE] Agree with the concept, but APDP is not adequate today.

[AGREE] A "Given!" But APDP is more than general acquisition—the software guys should be in the COMM/computer stall too.

• **Round II—Group Opinion: Agree**

[DISAGREE STRONGLY] Should be an AFMC program, best would be a DoD wide program (if it taught, what I think is the correct approach).

44. A SPO Training Program should address the training needs of the individuals managing the software acquisition process oriented to the specific procurement being performed (e.g. training individuals on the user requirements for the project).

• **Round I—Group Opinion: Agree Strongly**

[AGREE STRONGLY] Individual managers should be trained on each part of the acquisition process. Also, train the managers on the "user" not just the user requirements. Train on the user's mission; the organization of the user; the environment/circumstances driving their need for change.

[AGREE STRONGLY] What about the Air Force training program? Where do the SPO's learn?

[AGREE STRONGLY] Yes, train people in various test, analysis, CM, etc. methodologies and tools to be used to support acquisition.

[AGREE STRONGLY] You've got to know your SORDs which lead to SOWs and specs.

[AGREE] Also need training on the fundamentals of software acquisition and software development. Knowing "all" the user requirements won't help you acquire a quality software system.

- **Round II—Group Opinion: Agree Strongly**

[AGREE STRONGLY] A lot of the "user" orientation could be covered by moving/rotating software managers from the MAJCOMs through the acquisition centers.

[DISAGREE STRONGLY] Should be an AFMC program, best would be a DoD wide program (if it taught, what I think is the correct approach).

Integrated Software Management involves establishing and maintaining the project's defined software process and managing the software activities according to this defined software process and the special needs of the project. The project's defined software process integrates the management and technical processes as the basis for performing the project's activities. When project objectives are not being achieved, the managers know what actions need to be taken to correct the problem and reduce the likelihood of similar problems in the future. The organization provides support and historical data which the project uses to improve its software estimating, planning, and tracking process.

45. Is this Key Process Areas applicable to the proposed SAMF?

- **Round I—Group Opinion: Yes**

[YES] Even if not in the position to direct or influence a contractor's actions the program office should be in the position to understand and recommend.

[YES] Take the word "Software" out—then it is applicable.

- **Round II—Group Opinion: Yes**

46. Integrated Software Management implies a standard software acquisition process is defined at a higher organizational level than the SPO (i.e. Product Center).

- **Round I—Group Opinion: Agree**

[AGREE] Again, within a specific domain.

- **Round II—Group Opinion: Agree**

Software Product Engineering involves performing the technical activities to build and maintain the software system using appropriate state-of-the-practice tools and methods. The software requirements are identified, analyzed, refined, and documented. A software architecture and software designs are developed to implement the software requirements. The program code is developed to implement the software architecture and software designs. The software evaluation activities ensure that the software product to be delivered satisfies the specified requirements. A balance of flexibility and control is maintained to correct and revise the requirements, designs, and code to incorporate corrections and enhancements identified throughout the life cycle.

47. Is this Key Process Areas applicable to the proposed SAMF?

- **Round I—Group Opinion: Yes**

[YES] But needs refinement. Some of these tasks are monitoring tasks.

[YES] However, it must be applied as though "build and maintain" is changed to "acquire and support."

[YES] Again, take the word "software" out. This philosophy must cover the entire SPO if it is to be successful.

- **Round II—Group Opinion: Yes**

[NO] The SPO doesn't perform any of the following "SPE" activities: software requirements analysis, design, code, test, software maintenance. The SPO acts only on the management/procurement side of the house. The contractor does the analysis, design, code, and test, and the support agency whether a user or a support office does the maintenance.

[NO] Unless the SPO is developing its own software. Note: Here is where I need a SAMF/CMM translator to really give my best answer.

48. Software Product Engineering is a function of the contractor's software development organization and not a function of the SPO.

• **Round I—Group Opinion: Agree**

[AGREE] By the definition above, agree. However, under the DoD scheme for EMD the government becomes a participant via design reviews. Although not the primary actor, the government is a validator of activity along the way.

[DISAGREE] It's a function of both.

[DISAGREE] The SPO should be actively involved in measuring and evaluating the developer's product engineering process. Often the SPO should be actively involved in its definition.

[DISAGREE STRONGLY] The SPO should develop early prototypes for the user and use continuous step by step verification of existing processes and continuous process improvement.

• **Round II—Group Opinion: Agree**

Intergroup Coordination involves the disciplined interaction and coordination of the project groups with each other to address system-level issues and activities. System-level objectives and plans are established and used as the cornerstone for all project activities. The project groups participate, as appropriate, in defining the system requirements; establishing a system configuration; allocating requirements to hardware, software, firmware, and manual processes; monitoring and reviewing the hardware and software design and development; and managing and controlling changes to the system throughout the development effort. The technical working interfaces and interactions between groups are planned and managed to ensure the quality and integrity of the entire system.

49. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] But the contractor is the prime actor; the government is the monitor.

[YES] Is there a process to ensure that different parts of the SPO are working together?

[NO] This KPA doesn't work: Kill it; don't assume it can and must be fixed "at any cost." Fall back to MIL-STD-483, interface control working groups, advance system modeling or E-mail; create your own paradigm.

• **Round II—Group Opinion: Yes**

50. Intergroup Coordination is a very important attribute of both the software development organization and the SPO's software acquisition management organization.

• **Round I—Group Opinion: Agree Strongly**

[AGREE STRONGLY] Actually all other parts of the SPO (business management, etc.) as well as in plant representatives and customers.

[AGREE STRONGLY] This is an aspect of acquisition that is often missing.

[AGREE STRONGLY] To bad it doesn't happen.

[DISAGREE STRONGLY] This KPA doesn't work: Kill it; don't assume it can and must be fixed "at any cost." Fall back to MIL-STD-483, interface control working groups, advance system modeling or E-mail; create your own paradigm.

• **Round II—Group Opinion: Agree Strongly**

[DISAGREE STRONGLY] Good trick question. As I translate it to SAMF, you are asking if software acquisition management should coordinate with itself.

Peer Reviews involve a methodical examination of work products by the producer's peers to identify defects and areas where changes and improvements are needed. The specific work products that will undergo peer review are identified as part of the software planning activities. Peer reviews are performed following defined procedures. These procedures

cover preparing for the review, conducting the review, reporting the results of the review, and certifying review readiness/completion criteria. Actions identified in the review findings are documented and tracked until they are resolved.

51. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] This is highly resource intensive; schedule and cost savings process area that must be affected by the SPO, or, as in the case of SQA and IV&V, its loss will be more expensive than its cost.

[YES] However, the government's activity in this area is usually check-point, high level activities such as design reviews (PDR/CDR) and FCA/PCA vs in-line activities as part of the development process.

[YES] Although not exactly a "peer review," if the SPO's product is paper and correspondence, is there a planned way to coordinate it before release?

• **Round II—Group Opinion: No**

[NO] Who in the Air Force (with its accompanying rank structure) thinks that a peer review will be unbiased and honest? (My "peers," in experience and knowledge, out-rank me.)

[NO] The peer review process doesn't apply to the SPO management structure. The SPO doesn't develop software; peer reviews aren't applicable.

[NO] Not unless you increase numbers of software folks in each SPO. What's needed is central staff to do some reviewing.

52. Peer Reviews are a function of the contractor software development organization. The use of Peer Reviews is a decision made by the contractor and is not affected by the SPO software acquisition process.

• **Round I—Group Opinion: Disagree**

[DISAGREE] The SPO should insist on this activity via CPDP/SDP due to positive aspects that peer review/clean-room activities produce.

[DISAGREE STRONGLY] The SPO can mandate peer reviews by the contractor. Furthermore, SPOs can review deliverable products this way.

[DISAGREE STRONGLY] This is highly resource intensive; schedule and cost savings process area that must be affected by the SPO, or, as in the case of SQA and IV&V, its loss will be more expensive than its cost.

• **Round II—Group Opinion: Disagree**

[DISAGREE] Whether peer reviews are used by the contractor depends on the SPO's definition of "Sound software engineering practices." The SPO may require peer reviews or it may not, depending on the SPO software acquisition approach.

[DISAGREE] Need translator.

Process Measurement and Analysis involves taking measurements of the performance of the organization's standard software process (as instantiated by the projects), analyzing these measurements, and making adjustments to stabilize the process performance within acceptable limits. The process and the associated measurements are established as a baseline and are used to plan and control the process in quantitative terms.

53. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] However, the ability to influence the "making" of adjustments will depend on contractual constraints. We must understand the performance to assess risk.

• **Round II—Group Opinion: Yes**

[NO] It would be great but you would need much larger and stable staffing. Folks are stretched too thin right now.

54. Process Measurement and Analysis is an attribute that a SPO software acquisition organization should have at the mature level. The philosophy of

this attribute holds for the software acquisition process the same as for the software development process.

• **Round I—Group Opinion: Agree**

[AGREE STRONGLY] Acquisition process measurement techniques and tools, however, must be developed and maintained at levels higher than the SPO. This will enable metrics gathered to have value across projects and help SPOs decide which process model to use.

[DISAGREE] Sure, you need to take sanity checks along the way to insure that things are going well. But, this philosophy does not hold for the SPO since SPOs do not normally develop software products.

• **Round II—Group Opinion: Agree**

Quality Management involves defining software quality goals, establishing plans to achieve these goals, and monitoring and adjusting the software plans, activities, and quality goals to improve customer and end-user satisfaction. Quantitative product quality goals are established based on the needs of the organization, customer, and end users. Plans and process quality goals are established and the project's software process is specifically adjusted to achieve the product quality goals. The software activities and results are assessed against the quality objectives on a regular basis, and corrective actions are taken to bring forecasted process and product quality in line with the goals.

55. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] However, like other attributes above, this is more likely to be accomplished at design reviews and audits rather than continuous vigilance depends largely on the capability of the contractor.

[YES] However, again the action of "Quality Management" should be performed for the all processes at the SPO level, not just for software.

• **Round II—Group Opinion: Yes**

56. Quality Management implies quality software product and acquisition process goals are established and tracked. The philosophy of this attribute holds for the software acquisition process the same as for the software development process.

• **Round I—Group Opinion: Agree**

[AGREE STRONGLY] But who will set up the measures? (An ex-pilot-colonel?)

[AGREE] However, I'm confused as to the difference between this section and (31) to (32). Isn't this just the planning aspect of SQA? Why segregate them just because CMM does?

[DISAGREE STRONGLY] The application of quality management is not a knee jerk process nor should it be applied at the same levels in all cases. Lack of mature judgement and in-depth understanding of the juice vs. squeeze ratio has caused significant worker/management expectation problems.

• **Round II—Group Opinion: Agree**

[AGREE] I agree with the "Agree" comment in the first round.

Defect Prevention involves analyzing defects that were encountered in the past and taking action to prevent the injection of these types of defects in current and future project activities. Software activities are systematically reviewed by those who perform them to identify the defects that were encountered, to understand the root causes of the defects, and to determine the implications of the defects on future activities. Trends are analyzed to determine the kinds of defects that were encountered in the past. Defects which are likely to recur are identified, and specific actions are taken to prevent them.

57. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[YES] Trend analysis is usually a contractor function, but how does the SPO use the data?

• **Round II—Group Opinion: Yes**

[YES] Defect prevention should be practiced by the SPO in terms of "lessons learned" for the software acquisition process, not necessarily requirements/coding defects found per thousand lines of code.

[YES] It would be great but you would need much larger and stable staffing. Folks are stretched too thin right now.

58. Defect Prevention implies knowledge of when, where, and how defects are injected in the software product or process (either development or acquisition process). Furthermore, steps to prevent similar defects appearing are taken. The philosophy of this attribute holds for the software acquisition process the same as for the software development process.

• **Round I—Group Opinion: Agree**

[AGREE] This is also an important aspect of IV&V.

• **Round II—Group Opinion: Agree**

[AGREE STRONGLY] Another trick question. Also, need that translator to give answer.

Technology Innovation involves identifying, selecting, and evaluating new technologies, and incorporating the appropriate technologies into the organization's processes. A group acts as the focus for introducing technology innovations into the organization. By maintaining an awareness of software technology innovations throughout the world and systematically evaluating and experimenting with them, the organization selects appropriate technologies to improve its productivity and product quality. Pilot efforts are performed to assess new and unproven technologies before they are introduced across the organization. With appropriate sponsorship of the organization's management, the selected technologies are incorporated into the organization's process.

59. Is this Key Process Areas applicable to the proposed SAMF?

• **Round I—Group Opinion: Yes**

[NO OPINION] Depends. Not a high priority or every-project need.

• **Round II—Group Opinion: Yes**

[YES] But only in a limited way. Usually that's an XR function.

60. Technology Innovation implies a group is responsible for identifying, selecting, and evaluating new technologies for possible implementation into the acquisition management process. The implementation of such a group should be performed at a higher management level than the SPO (i.e. the product center).

• **Round I—Group Opinion: Agree**

[AGREE] "The implementation of such a group should be performed at a higher management level than the SPO (i.e. the Product Center)"—Why?

[AGREE] Maybe ever higher (SAF/DoD XRs). Could be a concept exploration task.

[DISAGREE] It should be done at both.

[DISAGREE] The best ideas come from the worker bees who are in the SPO's.

[DISAGREE] The KPA should be coordinated across AFMC with candidate technologies identified by the interaction between the laboratories, centers, SPOs and form a basis for an element of continuous process improvement. This should be monitored and reported on at all levels of the Air Force as a TQM initiative between users and AFMC.

[DISAGREE STRONGLY] While it may be appropriate for a product center to do this for some overall items, the SPO should do it for its functions and product center staff should assist. Help transfuse information across SPOs, etc.

• **Round II—Group Opinion: Agree**

61. Technology Innovation at the SPO level involves exploiting new technologies identified by the evaluation group to improve the software acquisition management process.

- **Round I—Group Opinion: Agree**

[AGREE] If technically and contractually appropriate.

[AGREE] Or, also, it may involve making the developer aware of technologies that they should be using.

- **Round II—Group Opinion: Agree**

Process Change Management involves defining process improvement goals and systematically identifying, evaluating, and implementing improvements to the organization's standard software process and the projects' defined software processes on a continuous basis. Appropriate training and incentive programs are established to allow and encourage all staff and managers to participate in these process improvement activities. Improvement opportunities are identified and evaluated for potential payback for the organization. Pilot efforts are performed to assess new and unproven process changes before they are introduced across the organization.

62. Is this Key Process Areas applicable to the proposed SAMF?

- **Round I—Group Opinion: Yes**

[YES] Yes but minor.

[NO] It should be combined with the Organizational Process Focus KPA.

- **Round II—Group Opinion: Yes**

[YES] It would be great but you would need much larger and stable staffing. Folks are stretched too thin right now.

63. Making changes for improvement to the software acquisition management process must be planned, implemented carefully, and measured to verify improvements were made.

- **Round I—Group Opinion: Agree**

[AGREE] But it is also important that the cost of making change does not outweigh the gains. You must have an efficient means of defining and re-defining process.

[DISAGREE] Overhead/bureaucratic waste and loss of mission focus result from every change; why make another area to manage these?

- **Round II—Group Opinion: Agree**

Comments on New Candidate Key Process Areas

64. Risk Management is required by DoD 5000.1 and 5000.2. It is an integral part of the overall SPO's acquisition management scheme. The characteristics of identifying and managing software cost, schedule, and performance risks should be a KPA in the proposed SAMF.

- **Round I—Group Opinion: Agree Strongly**

[AGREE STRONGLY] Also, most programs have the contractors' stating, "your government furnished property/equipment did not arrive in time, was not adequate, etc. to do our job." Thus, under risk, include the government furnished property issue.

[AGREE STRONGLY] In many ways, this area will drive others.

[DISAGREE STRONGLY] Risk is a part of life and so is loss; pseudo-sophistry in business management's application of statistical process controls lack the in-depth understanding of causal factors needed to estimate cost, schedule and performance relation-

ships and impacts. Burning more resources won't cure the problems caused by lack of rigor at the technical base.

- Round II—Group Opinion: Agree Strongly

65. Contract Management is a key function of any acquisition organization. This is the function of ensuring the contract meets both the user, maintainer, and SPO's acquisition requirements. The characteristic of timely and efficient contract modifications should be a KPA in the proposed SAMF.

- Round I—Group Opinion: Agree

[AGREE STRONGLY] For your purposes, the "subcontractor" management should be raised to overall "contract management." The expectations and overall management requirements are generally the same for SPO→Prime as for Prime→Subcontractor.

[AGREE] Only from the stand point that the software taskings have been properly defined, compliance documents have been properly tailored and all necessary deliverables have been requested.

[DISAGREE] Sounds like duplication of effort.

- Round II—Group Opinion: Agree

[DISAGREE STRONGLY] There is too much emphasis in making contract mods "timely" and "efficient" that we lose sight of the original goal. This is especially risky in the software arena. Better would be an informal tiered system in assigning priorities to software changes.

66. Data Management entails ensuring the correct software documentation is acquired from the contractor meeting the SPO's, user's, and maintainer's requirements. Furthermore, the amount of data should be periodically evaluated to ensure only the required amount is procured. The

characteristic of accurate and efficient data management should be a KPA in the proposed SAMF.

- Round I—Group Opinion: Agree Strongly

[AGREE STRONGLY] What is the documentation approval process? Defined? Is process defined to review requirements?

[DISAGREE] Data Management is largely subsumed by other areas. The documentation received from a developer is merely a "snapshot" of the information produced by one or more developer processes. The documentation itself has no value when taken out of context.

[DISAGREE] I see data management as a book-keeping function, tracking deliverables, which is an contractor function.

[DISAGREE STRONGLY] Data management does not ensure anything, other than tracking deliverable requirements list items on the contract. This responsibility is part of every maintainable product and should not be separate from contract management.

- Round II—Group Opinion: Agree Strongly

[AGREE STRONGLY] For example, what if a contractor folds? Without data, we would have a bunch of unmanageable software.

67. Software Supportability Planning is a key function of the software acquisition management process. Software support has been stated to account for approximately 60%-80% of the software life cycle costs. Furthermore, the decisions made during the early phases of the system life cycle have a great affect on the supportability of the software. The characteristics of continual software supportability planning throughout the software life cycle should be a KPA in the proposed SAMF.

- **Round I—Group Opinion: Agree Strongly**

[AGREE STRONGLY] Especially applicable within a SPO under IWSM—one should consider that under IWSM, the SPO may have some mature products it's supporting and some new developing (like we do).

[AGREE STRONGLY] Must have a CRWG and a CRLCMP (AFR 800-14). Work closely with the support and user communities to define the correct support concept.

[AGREE] A better more precise definition of software supportability is needed.

- **Round II—Group Opinion: Agree Strongly**

[AGREE STRONGLY] A system designed for maintainability is not only designed better, it will probably have a shorter development cycle. Software support planning must be a prime driver for the entire software team and not the concern of the logistics (ILSM) guy as one of ten sub-elements to support planning.

Appendix C

Software Acquisition Maturity Framework

◆ Acknowledgements:

This document is a direct product of the Air Force Institute of Technology Master's thesis "An Adaptation of the Software Engineering Institute's Capability Maturity Model to the Air Force Software Acquisition Process" (Dickerhoff and Sommers, 1992).

The authors would like to acknowledge that this document is an accumulation of the thoughts and ideas from many individuals. Primarily, the Software Acquisition Maturity Framework (SAMF) is an adaptation of the SEI's Capability Maturity Model (CMU/SEI-91-TR-24 and CMU/SEI-91-TR-25). Therefore, the general outline and a great deal of the text contained herein is a modification of or a direct quote from the SEI's CMM. The adaptation guidance came from Air Force software acquisition experts surveyed during the thesis effort as well as an in-depth review of the relevant literature. The thesis mentioned above should be consulted for any questions regarding the source material of any statements made within this document.

Definitions:

Organization — A group of programs within a specified application domain co-located at an Air Force Materiel Command product center.

Program Groups — A defined set of personnel involved in the system acquisition process (e.g. Using Command, PEO/PEM, SPO Project Team, SPO Functional Directorate, etc...).

Level 1—Initial

◆ **Key Process Area: Risk Management**

Risk Management is the organization's activities that identify, analyze, assess, and reduce program risk. The organization develops and follows a Risk Management Plan. Major areas of programmatic risk include cost, schedule, and performance.

Goals:

- ① The organization identifies, as early as possible, program areas that may cause problems in the program's completion and identifies the organization's ability to achieve its objectives.
- ② At a minimum, the organization controls risk concerning the areas of schedule, cost, and performance.

Commitment to Perform:

- ① The organization follows a documented plan for implementing a risk control process.
 - The plan includes direction for process planning, identification, assessment, analysis, and risk reduction.
 - The plan directs the organization to review the contractor's risk reduction plan.
 - The organization develops risk reduction measures and area report procedures to present to the appropriate decision authority during the milestone review.

Level 1—Initial

◆ **Key Process Area: Contract Management**

Contract Management is the organization's activities that initiate and maintain a contract that meets user requirements. The preparation for the Statement of Work is a vital part of the process. Furthermore, the contract must be periodically reviewed and updated to address any changes to the user's requirements. A process must be in place to allow the users to identify changes to requirements efficiently, to allow the program office to modify the contract to meet these changes, and to allow the contractor to easily adjust to the new requirements.

Goals

- ① The organization involves all necessary groups/individuals in the SOW development.
- ② The organization contracts for specific tasks recommended by the SOW preparation process.
- ③ The organization ensures that the contract, the SOW, and the Work Breakdown Structure accurately represent the user requirements.
- ④ The organization ensures that changes to the requirements reflect the changes in the contract vehicle.

Commitment to Perform:

- ① The organization follows a written plan for the preparation of the contract.
 - This plan requires:
 - That all groups affected by the SOW are represented on the SOW preparation team. (Some organizations that may be included on the team are the functional directorates within the program office as well as the user, the IV&V organization, and the test and evaluation organization).
 - The organization to document and to follow a SOW development schedule.
 - The organization to assemble an experienced acquisition personnel panel to review the SOW as it is developed.
 - The organization to periodically assemble experienced acquisition professionals to review the SOW development process.
- ② The organization follows a written policy for managing the requirements and associated contract changes.

Level 2—Repeatable

◆ **Key Process Area: Requirements Management:**

Requirements Management is the organization's activities that establish and maintain an understanding and agreement between the contractor and organization about the software requirements throughout the system's life cycle. The agreements cover both the technical requirements for the software and the nontechnical requirements, such as development schedule. The agreements form a basis for estimating, planning, and tracking the program's software activities.

Goals

- ① The organization ensures that the goals of the program office and contractor represent the user's needs.
- ② The organization ensures that the software system requirements provide a clear verifiable, and testable foundation for software development.
- ③ The organization ensures that the allocated requirements define the scope of the software effort.
- ④ The organization's plans, products, and activities represent the allocated requirements and their subsequent changes.
- ⑤ The organization establishes a formal training program that will teach individuals to management the requirements and how to make changes to them.

Commitment to Perform:

- ① The organization follows a written policy for managing the program requirements that determine and bound the software activities.
 - The policy requires:
 - The contractor to document and the SPO to approve the allocated requirements.
 - The users, support agencies, and other groups within the SPO to review and approve the allocated requirements.
 - The contractor's software plans, products, and activities reflect the changes in the allocated requirements.

Level 2—Repeatable

◆ **Key Process Area: Software Project Planning**

Software Project Planning is the organization's activities that verify the contractor's estimates for the work to be performed, that establish the necessary commitments, and that review the contractor's plan to perform the work. An organization plan must also be developed to establish the buyer's basis for managing the program.

Goals

- ① The organization documents its software acquisition management plan in the Computer Resources Life Cycle Management Plan. The plan defines the basis for managing the program
- ② The contractor develops a plan that appropriately and realistically covers the software activities and commitments. The buying agency reviews and approves this plan.
- ③ All affected groups and individuals understand and support the software estimates and plans.
- ④ The organization documents and verifies the contractor's software estimates.

Commitment to Perform:

- ① The organization designates a program office software manager to be responsible for reviewing commitments and approving the contractor's software development plan.
- ② The organization follows a written policy for planning a software acquisition program.

Level 2—Repeatable

◆ **Key Process Area: Software Project Tracking and Oversight**

Software Project Tracking and Oversight is the organization's activities that track and review the contractor's software accomplishments and results against the organization's documented estimates, commitments, and plans; and the organization adjusts these estimates, commitments and plans based on the contractor's *actual* accomplishments and results. The contractor documents a plan for the software effort. This plan is used as a basis for tracking software activities, communicating status, and revising plans. The organization conducts technical and management reviews with the contractor.

Goals

- ① The organization tracks the software program's actual results and performances against documented and approved plans.
- ② The organization takes corrective actions when the software program's actual results and performances deviate significantly from the plan.
- ③ The buyer monitors and adjusts the schedule of the development effort based on the actual data.

Commitment to Perform:

- ① The organization designates a program office software manager to be responsible for ensuring that the contractor is tracking its own results, activities, and performance.
- ② The organization follows a written policy for managing the contractor's tracking and oversight functions.
 - The policy requires:
 - The contractor to use a documented software development plan; the plan is maintained as the basis for tracking the software program.
 - The organization to track the contractor's work against authorized work tasks (SOW paragraphs).
 - The organization to provide the program office software manager appropriate visibility into software issues.
 - The organization to take corrective actions when the software development plan is not being achieved.
 - The organization to make software development plan changes with the involvement and approval of all affected groups. (The affected groups may contain the end user, the support organization, and different groups within the SPO.)

- ③ The organization follows a documented plan for contracting and using management indicators and metrics.

Level 2—Repeatable

◆ **Key Process Area: Software Subcontract Management**

Software Subcontract Management is the organization's activities that monitor the commitments between the prime and subcontractors and the efforts performed to achieve these commitments.

Goals

- ① The organization understands the contract commitments made by the prime and its subcontractors.
- ② The organization has visibility of the subcontractor's development effort.
- ③ The organization monitors the progress of the subcontracted efforts.

Commitment to Perform:

- ① The organization follows written policy for monitoring of the subcontractor's software efforts.
- ② The organization establishes a manager to be responsible for monitoring the subcontractor's effort and to advise the program manager of any major deviations from the documented plans.

Level 2—Repeatable

◆ **Key Process Area: Software Quality Assurance**

Software Quality Assurance is the organization's activities that ensure the contractor has adequately planned for and implements the review and audit of the development products and activities. An independent reporting path for the contractor's SQA personnel will be assured.

Goals

- ① The organization monitors the contractor's compliance with the applicable standards, procedures, and requirements.
- ② The organization reports compliance problems through an independent management path.
- ③ The organization ensures that senior management addresses noncompliance issues.

Commitment to Perform:

- ① The organization follows written policy for monitoring the contractor's SQA.

Level 2—Repeatable

◆ **Key Process Area: Software Configuration Management**

Software Configuration Management is the organization's activities that monitor and approve the selection of baselines, that control the changes in all baselines that are controlled by the organization, that monitor the changes to the baselines that are controlled by the developer, and that monitor the status of the program. Baselines changes (controlled by the buyer or the developer) must follow written procedures.

Goals

- ① The organization monitors the contractor's control and stabilization of the baselines used for planning, managing, and building the system.
- ② The organization ensures the integrity of the system's configuration over time.
- ③ The organization ensures the status and content of the software baselines are known.
- ④ The organization plans for the software baseline transition to the government.

Commitment to Perform:

- ① The organization follows written policy for contracting and implementing SCM.

Level 2—Repeatable

◆ **Key Process Area: Data Management**

Data Management is the organization's activities that identify the data items necessary for the system's life cycle and that ensure they are available to the using or supporting organization. The process should be recurring throughout the development cycle to ensure all appropriate data is acquired.

Goals

- ① The organization acquires or has access to all data and documentation necessary to support the system throughout its life cycle.
- ② The organization ensures that the data and documentation reflect the actual state of the product.

Commitment to Perform:

- ① Senior management understands and supports the importance of engineering data to the system's life cycle costs.
- ② The organization requires the contractor to present its data management plan to the program office for approval. (This plan should address the procedures for the use of such things as Software Development Folders.)
- ③ The organization follows a written policy for managing the engineering data on the program.
 - This policy requires the organization to:
 - Review its own data management process periodically to insure adequacy.
 - Give data management adequate resources to manage all the data necessary for program support.

Level 3—Defined

◆ **Key Process Area: Organization Process Focus**

Organization Process Focus is the organization's activities that develop and maintain an understanding of the organization's software acquisition process and that coordinate the activities that specify and improve these processes. A group, such as a Software Acquisition Process Group, acts as the focus for the software acquisition process activities in the organization. This group coordinates the programs' software acquisition process definition activities and the organization's long-term process improvement efforts.

Goals

- ① The organization understands their software acquisition process' strengths and weaknesses and establishes plans to systematically address the weaknesses.
- ② The organization establishes a group with appropriate knowledge, skills, and resources to define a standard software acquisition process for the organization.
- ③ The organization provides the resources and support needed to record and analyze the use of the organization's standard software acquisition process in order to improve it.

Commitment to Perform:

- ① Senior management sponsors the organization's activities for software acquisition process assessment, definition, and improvement.
- ② Senior management oversees the organization's activities for software process definition and improvement.

Level 3—Defined

► **Key Process Area: Organization Process Definition**

Organization Process Definition is the organization's activities that establish and maintain a standard software acquisition process for the organization along with related items. This process definition defines the essential process steps for the programs and establishes a common basis for measuring process performance across the organization.

Goals

- ① The organization defines a standard software acquisition process that maintains a basis for stabilizing, analyzing, and improving the software program's performance.
- ② The organization collects lessons learned from the programs and defines software acquisition sub-processes.

Commitment to Perform:

- ① The organization follows a written policy for governing the organization's and programs' software acquisition process definition.

Level 3—Defined

◆ **Key Process Area: Training Program**

Training Program is the organization's activities that identify the training needs of the organization, the programs, and the individuals and that develop and obtain training courses to address these needs. The training program ensures that the training needed to perform each organization's job functions is appropriate and is not circumvented inappropriately.

Goals

- ① The organization ensures that the staff and managers are trained and comprehend the organization's software acquisition process.
- ② The program office's staff and managers effectively use, or are prepared to use, the capabilities and features of the existing organization's work environment.
- ③ The organization provides the staff and managers with opportunities to improve their professional skills.
- ④ The organization manages the turnover of personnel and trains new individuals in the organization's software acquisition process.

Commitment to Perform:

- ① The organization follows a written policy for meeting its training needs.
 - The policy requires:
 - The organization to train individuals in the organization's software acquisition process and to train individuals on how the acquisition process fits into the entire program.
 - The organization to organize, identify, require and make available a set of training courses for each job function.
 - The organization to develop training courses within the organization, specifically covering the organization's software acquisition process.
 - All of the organization's staff and managers to receive a specified amount of training on the organization's software acquisition process.

Level 3—Defined

◆ **Key Process Area: Integrated Software Management**

Integrated Software Management is the organization's activities that establish and maintain the program's defined software process, that manage the software activities according to this process, and that manage the special needs of the program. The program's defined software acquisition process integrates the management and technical processes as the basis for performing the program's activities. When program objectives are not achieved, the program manager knows what actions to take to correct the problem and reduce the likelihood of similar problems occurring in the future. The organization provides support and historical data; the program uses the data to improve its software estimating, planning, and tracking process.

Goals

- ① The organization bases the plans and management of each program on the organization's standard software acquisition process.
- ② The organization uses technical and management data from past and current programs to effectively and efficiently estimate, plan, track, and re-plan the software acquisition programs.

Commitment to Perform:

- ① The organization follows a written policy for the software acquisition programs using the organization's standard software acquisition process.

Level 3—Defined

◆ **Key Process Area: Software Product Engineering**

Software Product Engineering is the organization's technical activities that acquire and support the software system using appropriate state-of-the-practice tools and methods. The organization monitors the contractor to ensure that software requirements are identified, analyzed, refined, and documented. The buyer reviews the software architecture and design to ensure it implements the contract requirements. Furthermore, the buyer ensures that the contractor adequately performs the required product assurance activities and that the software product satisfies the specified requirements.

Goals

- ① The organization addresses the process and the product's software engineering issues in the system requirements.
- ② The organization ensures that the software engineering activities are well-defined, integrated, and used consistently to acquire software systems.
- ③ The organization uses state-of-the-practice software engineering methods, as appropriate, to acquire and support the software system.
- ④ The organization systematically procures software engineering products that are consistent with each other and appropriate for acquiring and supporting the software system.

Commitment to Perform

- ① The organization follows a written policy that guides the software engineering activities.

Level 3—Defined

◆ **Key Process Area: Intergroup Coordination**

Intergroup Coordination is the organization's activities that coordinate group interaction within the program to address system-level issues and activities. The organization establishes and uses system-level objectives and plans as a cornerstone for all program activities. The program groups participate, as appropriate, in defining the system requirements; monitor and control the system configuration; review and approve the requirements allocated to hardware, software, firmware and manual processes; monitor and review the hardware and software design and development; and manage and control changes to the system requirements throughout the development effort. The organization plans and manages the technical working interfaces and interactions between groups to ensure the quality and integrity of the entire system.

Goals

- ① The staff and members understand and approve the program's technical goals and objectives.
- ② Personnel are aware of each group's assigned responsibilities and know the working interfaces between these groups.
- ③ The organization ensures that program groups are involved with intergroup activities and identifies, tracks, and addresses these intergroup activities.
- ④ The program groups work as a team.

Commitment to Perform

- ① The organization follows a written policy for enabling people from different groups to work together.

Level 3—Defined

◆ **Key Process Area: Software Supportability Planning**

Software Supportability Planning is the organization's activities that identify the software and system support concepts, and that plan for and acquire the appropriate support items. The organization should provide for a smooth transition from developmental support to operational support. A group will be appointed to advise the program manager on all software supportability related issues.

Goals

- ① The organization identifies all of the support concepts as early as possible.
- ② The organization provides for a smooth transition from the developmental environment to the operational and support environment.
- ③ The support agency has everything needed to support the product when it is fielded.
- ④ The organization charts a working group, such as the Computer Resources Working Group, to advise the program manager on issues concerning the software supportability.

Commitment to Perform:

- ① Senior management understands and supports the importance of software supportability to the system's life cycle costs.
- ② The organization follows a written policy for the software system support planning.
 - The policy requires the organization to:
 - Develop the software support concept in close coordination with the hardware support concept.
 - Document and review the software support concept periodically throughout the acquisition process.

Level 4—Managed

◆ **Key Process Area: Process Measurement and Analysis**

Process Measurement and Analysis is the organization's activities that measure the organization's performance on their standard software acquisition process (as instantiated by the programs), that analyze these measurements, and that make adjustments to stabilize the process performance within acceptable limits. The organization establishes the process and associated measurements as a baseline and uses it to plan and control the process in quantitative terms.

Goals

- ① The organization's standard software acquisition process is stable and is under statistical control.
- ② The organization identifies and controls the special causes of process variation (i.e., variation attributable to specific applications not inherent in the process).

Commitment to Perform:

- ① The organization follows a written policy for measuring and stabilizing its standard software acquisition process.
 - The policy requires the organization and programs to:
 - Implement a documented plan to bring the process under statistical control.
 - *Protect and limit access to sensitive data relating to individuals' performance.*
- ② The organization institutes a software acquisition metric program to provide the process data required for analysis.

Level 4—Managed

◆ Key Process Area: Quality Management

Quality Management is the organization's activities that define the software acquisition's quality goals, that establish plans to achieve these goals, and that monitor and adjust the software acquisition plans, activities, and quality goals to improve user satisfaction. The organization establishes quantitative product quality goals based on the needs of the organization and user. The organization establishes plans and process quality goals, and adjusts the program's software acquisition process to achieve the product quality goals. The organization assesses the software acquisition activities and results against the quality objectives on a regular basis, and takes corrective actions to bring forecasted process and product quality in line with the goals.

Goals

- ① The organization establishes measurable goals and priorities of the process for each program through interaction with the user, support organization, contractor, and all groups within the program office.
- ② The organization establishes measurable process quality goals for all groups involved in the software acquisition process.
- ③ The organization ensures that the software plans, designs, and acquisition process are adjusted to bring forecasted process and product quality in line with the goals.
- ④ The organization uses process measurements to manage the software acquisition program quantitatively.

Commitment to Perform:

- ① The organization follows a written policy for managing the quality of the software acquisition process on every program.
 - The policy requires:
 - The organization to document its commitment to improve the acquisition process quality continually.
 - The programs to define and quantitatively manage the measures from their defined software acquisition process.
 - The programs to define and monitor their process quality goal..
 - The organization to define the quality responsibilities for each group involved in the software acquisition process and to establish criteria to enable the group to check success in achieving quality.
 - Management to take action to prevent process defects from recurring whenever possible.

Level 5—Optimized

◆ **Key Process Area: Defect Prevention**

Defect Prevention is the organization's activities that analyze defects encountered in the past and that prevent the injection of these defects in current and future program activities. Software acquisition activities are systematically reviewed by those who performed them to identify the defects, to understand the root causes of the defects, and to determine the implications of the defects on future activities. Defects that are likely to recur are identified and specific actions are taken to prevent them.

Goals

- ① The organization identifies and eliminates the sources of process defects that are inherent or occur repeatedly in the software acquisition process activities.

Commitment to Perform:

- ① The organization follows a written policy for governing defect prevention activities.
 - The policy requires that defect prevention activities are:
 - Implemented across the organization to improve the software acquisition process.
 - Included in each program's software management plan.
- ② Management supports and participates in defect prevention activities.
 - Specifically, management:
 - Establishes long-term plans and commitments for defect prevention funding and staffing.
 - Allocates resources needed for the defect prevention activities.
 - Handles management actions that resulted from defect prevention activities.
 - Reviews the defect prevention results to ensure the activities are effective and resolve management issues.

Level 5—Optimized

◆ **Key Process Area: Technology Innovation**

Technology Innovation is the organization's activities that identify, select, and evaluate technologies, and that incorporate the appropriate technologies into the organization's process and into the products the organization acquires. A group acts as the focus for introducing technology innovations into the organization. By maintaining an awareness of software development and software acquisition technology innovations throughout the industry and systematically evaluating and experimenting with them, the organization selects appropriate technologies to improve its process. The organization pilots efforts to assess new and unproven technologies before they are introduced across the organization. With appropriate sponsorship of management, the organization incorporates the selected technologies into the organization's process.

Goals

- ① The organization allows the programs to capitalize on the best available technologies in the industry.
- ② Selection and transfer of new technology into the organization is orderly and thorough.
- ③ Technology innovations are tied to quality and productivity improvements of the organization's standard software acquisition process.
- ④ An organizational level group is responsible for collecting and disseminating information on new technologies.

Commitment to Perform:

- ① The organization follows a written policy for improving its technology capability.
 - This policy requires the organization to:
 - Establish technology innovation objectives in its strategic and operating plans.
 - Implement a documented plan to address the technology innovation objectives.
- ② Senior management sponsors the organization's technology innovation activities.
 - Specifically, senior management:
 - Defines a strategy for technology innovation.
 - Coordinates with the programs to define their goals and methods to accomplish the organization's strategy.
 - Makes a commitment to the technology innovation effort that is visible to organization's staff and managers.
 - Establishes long-term plans and commitments for funding, staffing, and other resources.

③ Senior management oversees the organization's technology innovation activities.

- Specifically, senior management:

- Helps to establish technology innovation policies and reviews and approves these policies.

- Allocates resources for technology innovation activities.

- Helps relate organizational strategies and objectives to technology innovation strategies.

- Participates in establishing the technology innovation plans.

Level 5—Optimized

◆ **Key Process Area: Process Change Management**

Process Change Management is the organization's activities that define process improvement goals and systematically identify, evaluate, and implement improvements to the organization's standard software acquisition process and the programs' defined software acquisition process on a continuous basis. The organization establishes appropriate training and incentive programs and encourages all staff and managers to participate in these process improvement activities. The organization identifies and evaluates the improvement opportunities for potential payback. The organization pilots efforts to assess new and unproven process changes before they are introduced across the organization.

Goals

- ① The organization actively involves the staff and managers in setting quantitative, measurable improvement goals and in improving the software acquisition process.
- ② The organization ensures that their standard software acquisition process and the programs' defined software acquisition process continually improve.
- ③ The organization's staff and managers use the evolving software acquisition process and their supporting tools and methods properly and effectively.

Commitment to Perform:

- ① The organization follows a written policy for implementing software acquisition process improvements.
 - The policy requires:
 - The organization to have quantitative, measurable goals for the software acquisition process improvement and to track performance against these goals.
 - The organization to improve their acquisition process and to increase the quality of the product acquired.
 - The organization's staff and managers to participate in improving the software acquisition process.
 - The organization to establish comprehensive programs to improve skills, enhance job satisfaction, and to ensure appropriate job assignments.
- ② Senior management oversees the organization's activities for software acquisition process improvement.
 - Specifically, senior management:
 - Establishes the organization's long-term goals and plans for process improvement.

- Allocates resources for process improvement activities.
- Coordinates with the software acquisition managers to ensure they have reasonable and aggressive process improvement goals and effective process improvement plans to meet these goals.
- Monitors process improvement performance against goals.
- Maintains a consistent priority focus on process improvement in the face of acquisition crises.
- Ensures that process improvement issues are promptly resolved.

Appendix D

Acronym List

ACAT	Acquisition Category
AFCAC	Air Force Computer Acquisition Center
AFIT	Air Force Institute of Technology
AFLC	Air Force Logistics Command
AFMC	Air Force Material Command
AFOTEC	Air Force Operational Test and Evaluation Center
AFSC	Air Force Systems Command
ALC	Air Logistic Center
APDP	Acquisition Professional Development Program
ASC	Aeronautical Systems Center
CCP	Contract Change Proposal
CM	Configuration Management
CMM	Capability Maturity Model
CRAC	Computer Resources Acquisition Course
CRLCMP	Computer Resources Life-Cycle Management Plan
CRWG	Computer Resources Working Group
CSC	Computer Software Component
CSCI	Computer Software Configuration Item
CSU	Computer Software Unit
DAB	Defense Acquisition Board
DAE	Defense Acquisition Executive
DoD	Department of Defense
DoDD	DoD Directive
DoDI	DoD Instruction
DPRO	Defense Plant Representative Office
DSMC	Defense Systems Management College
ECP	Engineering Change Proposal
ESC	Electronic Systems Center
FRDC	Federal Research Development Center
GFE	Government Furnished Equipment

HSC	Human Systems Center
IG	Inspector General
IRS	Interface Requirements Specification
ISO	International Standards Organization
IWSM	Integrated Weapon Systems Management
KPA	Key Process Area
MAJCOM	Major Command
MCCR	Mission Critical Computer Resources
OPF	Operational Flight Program
PEO	Program Executive Officer
PM	Program Manager
RFP	Request for Proposal
SAE	Service Acquisition Executive
SAMF	Software Acquisition Maturity Framework
SAMM	Software Acquisition Maturity Model
SAPG	Software Acquisition Process Group
SCE	<i>Software Capability Evaluation</i>
SDCCR	Software Development Capability Capacity Review
SDF	Software Development File
SDP	Software Development Plan
SEI	Software Engineering Institute
SEMP	Systems Engineering Master Plan
SMC	Space and Missile Center
SOW	Statement of Work
SPDP	Software Professional Development Program
SPO	Systems Program Office
SQA	Software Quality Assurance
SRMP	Software Risk Management Plan
SRS	Software Requirements Specification
TEMP	Test and Evaluation Master Plan
TQM	Total Quality Management
WBS	Work Breakdown Structure

Bibliography

- Abdel-Hamid, Tarek and Stuart E. Madnick. *Software Project Dynamics: An Integrated Approach*. Englewood Cliff NJ: Prentice Hall, Inc, 1991.
- Baumert, John. "New SEI Maturity Model Targets Key Practices," *IEEE Software*, 8: 78-79 (November 1991).
- Baylor, Paul. CSCE 595 class lecture, Software Generation and Maintenance. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, June 1992.
- Beer, Stafford. *Decision and Control: The Meaning of Operational Research and Management Cybernetics*. New York: John Wiley and Sons, 1978.
- Bersoff, Edward H. et al. *Software Configuration Management: An Investment in Product Integrity*. Englewood Cliffs NJ: Prentice Hall Inc, 1980.
- Boehm, Barry W. *Software Engineering Economics*. Englewood Cliff NJ: Prentice Hall, Inc, 1981.
- Bollinger, Terry B. and Clement McGowan. "A Critical Look at Software Capability Evaluations," *IEEE Software*, 8: 25-41 (July 1991).
- Brown, Bernice B. *Delphi Process: A Methodology Used for the Elicitation of Opinions of Experts*. Santa Monica CA: Rand Corporation, September 1968 (AD-675 981).
- The Capability Maturity Model (Tutorial)*. Pittsburgh: Software Engineering Institute, Carnegie Mellon University, 24 January 1991.
- Card, David. "Understanding Process Improvement," *IEEE Software*, 8: 102-103 (July 1991).
- Cochrane, Charles B. "Defense Acquisition Policy: A New Set of Directives for 'A Disciplined Management Approach'," *Program Manager*. 29-34 (May-June 1991).
- Corbi, T.A. "Program Understanding: Challenge for the 1990s," *IBM Systems Journal*, 28: 294-306 (1989).
- Curtis, William, Director Software Process Program (Software Engineering Institute). "The Superior Software Organization: Extending the Capability Maturity Model." Addressed at CMM/SEPG Workshop Review. Tysons Corners VA, 29 May 1992.
- Dalkey, Norman C. *Delphi*. Santa Monica CA: Rand Corporation, October 1967 (AD-660 554).
- Dalkey, Norman C. et al. *The Delphi Method, IV: Effect of Percentile Feedback and Feed-In of Relevant Facts*. Contract F44620-67-C-0045. Santa Monica CA: Rand Corporation, March 1970 (AD-702 790).
- Deming, W.E. *Out of the Crisis*. Cambridge MA: MIT Center for Advanced Engineering Study, 1982.
- Department of the Air Force. *Air Force Mission Needs and Operational Requirements Process*. AFR 57-1. Washington: HQ USAF, 8 November 1992.

- Department of Defense. *Defense Acquisition*. DOD Directive 5000.1. Washington: Government Printing Office, 23 February 1991.
- _____. *Defense Acquisition Management Policies and Procedures*. DOD Instruction 5000.2. Washington: Government Printing Office, 23 February 1991.
- _____. *Defense System Software Development*. DoD-STD-2167A. Washington: Government Printing Office, 29 February 1988.
- _____. *Defense Systems Software Quality Program*. DoD-STD-2168. Washington: Government Printing Office, 1 August 1979.
- _____. *Software Technology Strategy*. Draft Report. Washington: Director of Defense Research and Engineering, December 1991.
- _____. *Technical Reviews and Audits for Systems, Equipment, and Computer Software*. MIL-STD-1521B. Washington: Government Printing Office, 4 June 1985.
- Dorling, Alec. *The Need and Requirements for a Software Process Assessment Standard*. Process Management Study Report for International Standards Organization. ISO/IEC JTC1/SC7/WG7/SG1. Cranfield IT Institute and UK Ministry of Defence, 7 March 1992.
- Emory, William C. and Donald R. Cooper. *Business Research Methods* (fourth edition). Boston: Richard D. Irwin, Inc, 1991.
- Evaluation Team Training: Participant's Handbook*. Pittsburg: Software Engineering Institute, Carnegie Mellon University, 24 January 1991.
- Humphrey, Watts S. "Characterizing the Software Process: A Maturity Framework," *IEEE Software*, 15: 73-79 (March 1988).
- _____. *Managing the Software Process*. Reading MA: Addison-Wesley Publishing Company, 1990.
- _____. et al. "Software Process Improvement at Hughes Aircraft," *IEEE Software*, 8: 11-23 (July 1991).
- _____. and W.L. Sweet. *A Method for Assessing the Software Engineering Capability of Contractors*. Technical Report ESD-TR-87-186, preliminary version. Pittsburgh: Software Engineering Institute, Carnegie Mellon University, September 1987.
- Linstone, Harold A. and Murray Turoff, ed. *The Delphi Method: Techniques and Applications*. Reading MA: Addison-Wesley Publishing Company, 1975.
- Marciniak, John J. and Donald J. Reifer. *Software Acquisition Management: Managing the Acquisition of Custom Software Systems*. New York: John Wiley and Sons, Inc, 1990.
- Martino, Joseph P. *Technological Forecasting for Decisionmaking*. New York: Elsevier, 1975.
- McKissik, John Jr. and Robert A. Price. "The Software Development Notebook - A Proven Technique." Proceedings 1979 Annual Reliability and Maintainability Symposium, 1979
- Mead, E.H., Member of the Technical Staff. Personal interview. The Software Engineering Institute, Carnegie Mellon University, Pittsburgh PA, 7 November 1991.

- _____. "Software Acquisition Maturity Model." Unpublished draft Task Plan. The Software Engineering Institute, Carnegie Mellon University, Pittsburgh PA, 19 July 1991.
- Millett, Stephen M. and Edward J. Honton. *A Manager's Guide to Technology Forecasting and Strategy Analysis Methods*. Columbus OH: Battelle Press, 1991.
- Osterweil, Leon, Professor at the University of California at Irvine. "Software Process: A State of the Art Report." Addressed at 14th International Conference on Software Engineering. Melbourne, Australia, 13 May 1992.
- Paulk, Mark C. et al. *Capability Maturity Model for Software*. Pittsburgh: Software Engineering Institute, Carnegie Mellon University, August 1991.
- Peschke, Richard E. and Marcus L. Sherrill. *Management Cybernetics: An Application to the Development of a Conceptual Model of the Software Acquisition Management Discipline*. MS thesis, AFIT/GLM/LSSR/2-79B. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, October 1979 (AD-A076946).
- Randolph, General Bernard P. Presentation given at Electronic Systems Division, Hanscom AFB MA, 1989.
- Software Capability Evaluation Overview: Participant's Handbook*. Pittsburgh: Software Engineering Institute, Carnegie Mellon University, 24 January 1991.
- Stein, Jess and Leonore C. Hauck ed. *The Random House College Dictionary* (revised edition). New York: Random House, Inc, 1980.
- Topper, Andrew and Paul Forgensen. "More Than One Way To Measure Process Maturity," *IEEE Software*, 8: 9-10 (November 1991).
- Weber, Charles V. et al. *Key Practices of the Capability Maturity Model*. Pittsburgh: Software Engineering Institute, Carnegie Mellon University, August 1991.

Vita

(William G. Dickerhoff)

Captain William G. Dickerhoff, Jr. was born on 18 October 1965 in Canandaigua, New York. He graduated from Battle Ground Academy high school in Franklin, Tennessee in 1983. Captain Dickerhoff then attended Vanderbilt University in Nashville, Tennessee where he earned a Bachelor of Engineering degree in Electrical Engineering in May of 1987. He received his commission as an officer in the United States Air Force and reported to his first duty station: Edwards AFB, California. At Edwards, Captain Dickerhoff was assigned the duties of Computer Engineer and Software Evaluator for the 31st Test and Evaluation Squadron (SAC). His responsibilities included evaluating over seven hundred fifty thousand lines of software code as well as its related computer equipment for a highly-classified Air Force acquisition program. Captain Dickerhoff entered the School of Systems and Logistics in May of 1991 in pursuit of a Master of Science in Software Systems Management.

Permanent Address:

404 Perkins Drive

Franklin, TN 37064

Vita

(William J. Sommers)

Captain William J. Sommers was born on 6 June 1964 in Nashua, New Hampshire. He graduated from Kwajalein Jr./Sr. High School in the Marshall Islands in 1982. He attended Norwich University in Northfield, Vermont earning a Bachelor of Science degree in Computer Science Engineering in May 1987. Upon graduation, he received his commission in the United States Air Force through the reserve officers training corps. His first duty assignment was to the Joint STARS program office, Hanscom AFB, MA. There he served in the Test, Engineering, and Projects directorates respectively during his four year tour. His responsibilities included obtaining test resources, monitoring the software development for the electronic warfare suite, managing the government's review of the radar software allocated specification, and managing the government's review of all software test plans and procedures for the formal verification of 760K lines of code. Capt Sommers entered the School of Systems and Logistics in May of 1991 in pursuit of a Master of Science in Software Systems Management.

Permanent Address:

RFD #1 Box 1405

North Vassalboro, ME 04962

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 the 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 Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE December 1992	3. REPORT TYPE AND DATES COVERED Master's Thesis		
4. TITLE AND SUBTITLE THE ADAPTATION OF THE SEI'S CAPABILITY MATURITY MODEL TO THE AIR FORCE SOFTWARE ACQUISITION MANAGEMENT PROCESS		5. FUNDING NUMBERS		
6. AUTHOR(S) William G. Dickerhoff Jr., Captain, USAF William J. Sommers, Captain, USAF				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology, WPAFB OH 45433-6583		8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GSS/LSY/92D-3		
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSORING / MONITORING AGENCY REPORT NUMBER		
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution unlimited		12b. DISTRIBUTION CODE		
13. ABSTRACT (Maximum 200 words) This study develops an Air Force Software Acquisition Maturity Framework (SAMF) by adapting the Software Engineering Institute's (SEI) Capability Maturity Model (CMM) to the Air Force software acquisition process. The SAMF's purpose is to provide the Air Force Materiel Command's product centers and program offices with criterion to assess their software acquisition maturity in a similar fashion as the SEI's CMM provides companies a benchmark to measure their organization's software production maturity. The research was accomplished through a combination of information gathering techniques and data analysis. A literature search of documentation, both within and external to the Department of Defense, identified several components of the Air Force software acquisition process. A Delphi survey collected several software acquisition experts' opinions and recommendations on the research. Based on the information gathered, the authors determined the SAMF to have the identical structure as the CMM. Twenty-one software acquisition process Key Process Areas (KPA) were identified. Definition of these KPAs was based on the research data gathered. The study produced the initial framework for an Air Force Software Acquisition Maturity Model.				
14. SUBJECT TERMS Delphi Techniques, Computer Programs, Acquisition, Air Force Procurement, Process Model, Systems Management		15. NUMBER OF PAGES 196		16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

AFIT RESEARCH ASSESSMENT

The purpose of this questionnaire is to determine the potential for current and future applications of AFIT thesis research. Please return completed questionnaires to: AFIT/LSC, Wright-Patterson AFB OH 45433-9905.

1. Did this research contribute to a current research project?

a. Yes

b. No

2. Do you believe this research topic is significant enough that it would have been researched (or contracted) by your organization or another agency if AFIT had not researched it?

a. Yes

b. No

3. The benefits of AFIT research can often be expressed by the equivalent value that your agency received by virtue of AFIT performing the research. Please estimate what this research would have cost in terms of manpower and/or dollars if it had been accomplished under contract or if it had been done in-house.

Man Years _____

\$ _____

4. Often it is not possible to attach equivalent dollar values to research, although the results of the research may, in fact, be important. Whether or not you were able to establish an equivalent value for this research (3, above) what is your estimate of its significance?

a. Highly
Significant

b. Significant

c. Slightly
Significant

d. Of No
Significance

5. Comments

Name and Grade

Organization

Position or Title

Address