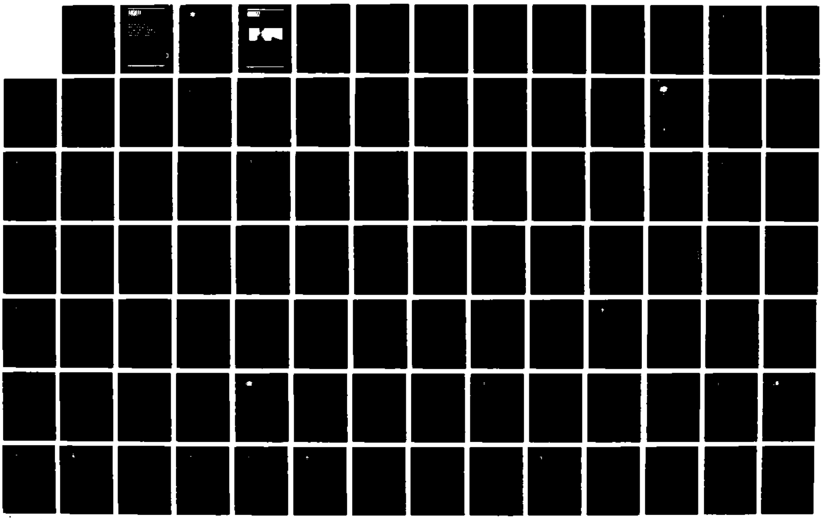
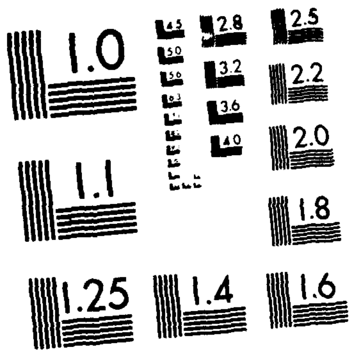
AD-A191 875 RISK ASSESSMENT METHODOLOGY FOR SOFTWARE SUPPORTABILITY (RAMSS): PILOT EV..(U) BDM CORP ALBUQUERQUE NM D E PEERCY ET AL. 14 APR 86 BDM/ABQ-86-0360-TR 1/3

UNCLASSIFIED F29601-85-C-0058 F/G 12/5 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963 A

AD-A191 875

THE BDM CORPORATION

1801 Randolph Road, S.E., Albuquerque, NM 87106 • (505) 848-5000

Risk Assessment Methodology for Software Supportability (RAMSS): Pilot Evaluation Results and Methodology Refinement

April 14, 1986

BDM/ABQ-86-0360-TR

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

AD-A191875

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS N/A
2a. SECURITY CLASSIFICA[redacted]RITY N/A	3. DISTRIBUTION/AVAILABILITY OF REPORT Unlimited
2b. DECLASSIFICATION/DO[redacted]G SCHEDULE N/A	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) BDM/ABQ-86-0360-TR	5. MONITORING ORGANIZATION REPORT NUMBER(S) N/A
6a. NAME OF PERFORMING ORGANIZATION THE BDM CORPORATION 6b. OFFICE SYMBOL *(If applicable)*	7a. NAME OF MONITORING ORGANIZATION Air Force Operational Test and Evaluation Center
6c. ADDRESS *(City, State and ZIP Code)* 1801 Randolph Rd SE Albuquerque, New Mexico 87106	7b. ADDRESS *(City, State and ZIP Code)* Kirtland Air Force Base, New Mexico 87117-7001
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Same as 7a 8b. OFFICE SYMBOL *(If applicable)* RMC	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F29601-85-C-0058
8c. ADDRESS *(City, State and ZIP Code)* Same as 7b	10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
SS 412/01		7.3	

11. TITLE *(Include Security Classification)*
Risk Assessment Methodology for Software...

12. PERSONAL AUTHOR(S)
D. E. Peercy, W. F. Huebner, M. D. Estill, K. L. Shaw

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
Technical	FROM 10/24/85 TO 4/14/86	1986 April 14	200

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUB. GR.	

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

The Risk Assessment Methodology for Software Supportability (RAMSS) was developed for the Air Force Operational Test and Evaluation Center (AFOTEC) to provide decision makers with information from which the risk to the Air Force of supporting a software system can be derived. The purpose of this report is to document a pilot application of the RAMSS and to describe the refinements and procedures necessary for AFOTEC to apply the RAMSS to future programs. The system chosen by AFOTEC for the pilot application was the Additionally, this report includes an overview briefing of the RAMSS, and an Evaluator's Guide for use by AFOTEC personnel when applying the RAMSS to other systems.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. X DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE NUMBER (Include Area Code)	22c. OFFICE SYMBOL
Captain Eric H. Tomlin	(505) 846-1381	AFOTEC/LG5T

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

BDM

THE BDM CORPORATION

1801 Randolph Road, S.E., Albuquerque, NM 87106 • (505) 848-5000

ASSE

PILOT EVA

METH

Apr

BDM/ABO-86-0360-TR

DISTRIBUTION: UNLIMITED

FOREWORD

a. This technical report, BDM/ABQ-86-0360-TR, is submitted by The BDM Corporation, 1801 Randolph Road SE, Albuquerque, New Mexico 87106 to the Air Force Operational Test and Evaluation Center, Kirtland Air Force Base, Albuquerque, New Mexico 87117-7001. This submission is in compliance with the requirements of paragraph 7.3 of Subtask Statement 412/01, titled "Risk Assessment Methodology for Software Supportability (RAMSS): Pilot Evaluation Results and Methodology Refinement."

b. This report is the result of effort by Dr. David Peercy (Technical Leader), Mr. Walter Huebner, Jr. (Task Leader), Mr. Donan Estill, and Ms. Kelley Shaw, of The BDM Corporation. The primary Subtask Statement Project Officer is Capt. Eric H. Tomlin (AFOTEC/LG5T); the alternate Subtask Statement Project Officer is Maj. Gary R. Horlbeck (AFOTEC/LG5T)/

Reviewed and approved by:

Walter F. Huebner
Program Manager

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Concluded)

LIST OF FIGURES

LIST OF FIGURES (Concluded)

LIST OF TABLES

I. Introduction

SECTION I

INTRODUCTION

1.1 PURPOSE OF REPORT.

a. The purposes of this report are to document:

(1) The pilot application of the risk assessment methodology for software supportability (RAMSS) to an on-going Air Force Operational Test and Evaluation Center (AFOTEC) program

(2) The refinements to the risk assessment methodology based upon the pilot application

(3) The procedures which AFOTEC will use in the application of the RAMSS to future programs.

b. It is intended that, to the maximum extent possible, the procedures discussed in this report will provide AFOTEC with a completed methodology which can be validated and applied to future software evaluations independent of contractor support.

1.2 OBJECTIVES OF RAMSS.

1.2.1 Overview of Objectives.

a. AFOTEC has the responsibility for conducting operational test and evaluation (OT&E) of assets entering the Air Force inventory. AFOTEC has developed and implemented various software OT&E methodologies. These methods have matured and have become the Air Force standard for evaluating software supportability. Each of these developed methods evaluates specific characteristics of the supportability aspects of delivered software and software support resources. These stand-alone evaluations provide AFOTEC with information to identify particular software supportability deficiencies, but do not identify overall risk associated with contractor or military ownership and organic maintenance of contractor-delivered software.

b. The development of the RAMSS has resulted from AFOTEC's concern about the need for a risk assessment method which provides software testers with areas which require testing emphasis and decision makers with an assessment of the software supportability risk. The objectives of the RAMSS can be classified as both programmatic and technical. In particular, the programmatic objectives are to provide a method which allows:

(1) Early planning and trade-off studies for software support resource requirements

(2) Early visibility of requirements for expected software support actions

(3) Early view of potential software support management problems

(4) Capability to trace software supportability risk measures throughout the system life cycle.

c. The technical objectives, which complement the program objectives, are that the method should:

(1) Have a technical depth and resulting format appropriate to adequately assist decision makers

(2) Integrate at least the current AFOTEC evaluation methodologies

(3) Have enough accuracy and repeatability to warrant confidence in the results

(4) Be based upon a sound theoretical software and risk assessment foundation

(5) Allow for determination of what acceptable level of risk means depending upon the identity of the risk agent and the software supportability requirements.

d. The following subsections will give the reader a brief background review of the RAMSS development and discuss the major elements of the method.

1.2.2 Concept Development.

a. Since 1982, AFOTEC has been analyzing the problem of how to assess the risk to the Air Force of supporting software acquired for weapon systems. A concept for computer resources risk assessment during operational test and evaluation was proposed in 1983 (reference 1.4.11). Several issues evolved from this proposal. First, the assessed risk should reflect software supportability impact upon the system at a level appropriate for AFOTEC reporting requirements. Second, supportability is a concern for both the user and the supporter. Any defined risk of software supportability should reflect some aspect of user risk and supporter risk. Third, current AFOTEC methods of evaluating software supportability should be integrated into the risk assessment method. Also, the risk assessment method should be adaptable to include other AFOTEC concerns such as software maturity and software reliability.

b. This initial concept proposal provided AFOTEC with justification to study the feasibility of developing and implementing a risk assessment methodology for software supportability (RAMSS). The approach for this study (references 1.4.2, 1.4.3, 1.4.4) included:

(1) Literature review and assemblage of a data base of relevant tools, techniques and methods

(2) Analysis of relevant tools, techniques, and methods for feasibility of application to AFOTEC's needs

(3) Development of a framework for assessing software supportability risk along with a preliminary set of risk measures.

c. The primary conclusion from this feasibility study was that a RAMSS could be developed based upon the framework derived as part of the study. However, there were still several technical issues which needed to be resolved. Of these issues, the major one concerned the need to establish a baseline against which to measure risk. Since risk was defined (for this study) as "the potential for realization of unwanted, negative consequences of an event," it was necessary to have a baseline of software support activities in order to tell when a consequence may be negative. This baseline, called an historical maintenance profile, reflects how software support resources are being used to perform the software support activities. Given this information, the framework recommended by the feasibility study could be used to compute measures of risk and incorporate the issues proposed in 1983.

1.2.3 Methodology Requirements (Inputs). Figure 1-1 illustrates interfaces with the RAMSS. The inputs consist of:

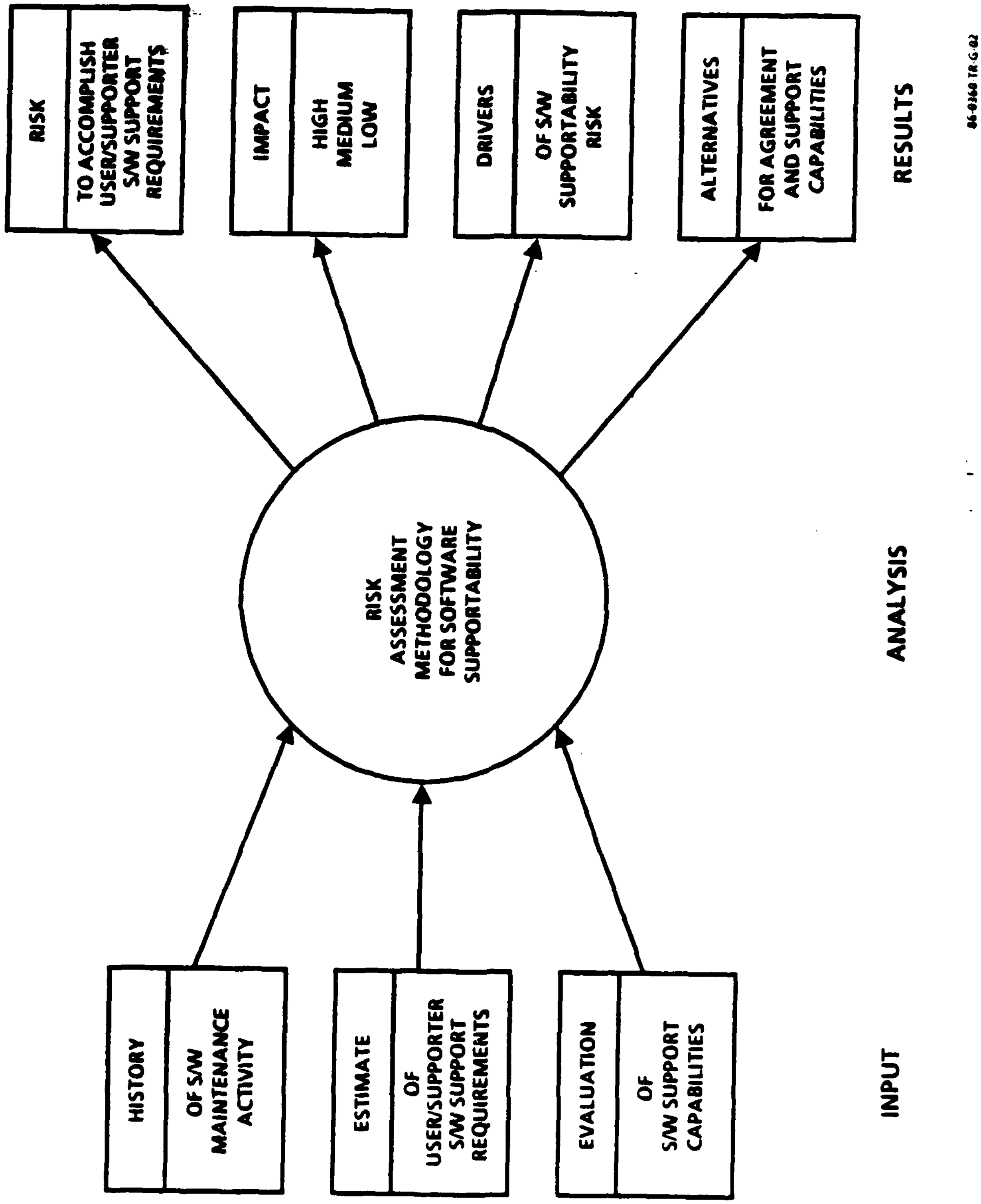


Figure I-1. High Level View of Risk Assessment Methodology for Software Supportability

(1) The historical profile of software maintenance activity and software supportability evaluations

(2) A user/supporter baseline estimate of planned software maintenance changes and support resource requirements for the software system being evaluated

(3) An evaluation of software support capabilities using current AFOTEC methods.

1.2.4 Methodology Analysis. The RAMSS inputs are combined and analyzed, and measures of risk are computed for the system being evaluated.

1.2.5 Methodology Benefits (Results).

a. The major results of the RAMSS are also illustrated in figure 1-1. These results include:

(1) The software supportability risk measure which quantifies the probability of the user/supporter baseline estimate not being accomplished with current software support capabilities

(2) The capability to identify the impact of the software supportability risk as high, medium, or low

(3) The identification of the drivers of the software supportability risk

(4) The projection of alternative choices for risk reduction (for instance, by improving certain aspects of current or projected software support capabilities).

b. With this information, the decision maker can assess the effect of software supportability upon system suitability and effectiveness. In addition, detailed data are available to help answer specific questions such as why particular areas of software supportability are drivers and how the software supportability risk can be reduced to an acceptable level.

1.2.6 Baseline Definition and Application.

a. As discussed above, a key element to the risk assessment process is recognition that software supportability is important both to the user and to the supporter of the software. Therefore any risk assessment methodology which ignores the interests of one of these parties may estimate a risk that is unacceptable to the other. In an attempt to bridge this gap, the RAMSS input requires a User/Supporter Baseline Estimate be established, and that the evaluations of software support capabilities be made against that baseline.

b. The User/Supporter Baseline Estimate uses inputs from both the user (using command) and the supporter (supporting command). The estimate includes an understanding of the software block release cycle, projected software support personnel (numbers and types), and anticipated software change request activity for each block release. Details of the User/Supporter Baseline Estimate are contained in section II of this report.

c. The current AFOTEC methods for evaluating software supportability do not consider in a direct manner the effect of an estimated baseline. The establishment of an estimated baseline is critical to risk assessment because it (1) provides a means to judge how well the measured risk agrees with the estimated risk (which is, in some sense, "acceptable" to both the user and the supporter), and (2) quantifies the options required to lower the measured and acceptable risks (a desired result of the risk assessment process).

This report documents what steps should be taken during the evaluation of a system's software supportability to ensure that the User/Supporter Baseline Estimate is accounted for in the risk assessment process.

1.3 GENERAL ORGANIZATION OF REPORT.

The remainder of this report is organized into three additional sections, plus a set of appendices which provide useful support information for the RAMSS. Report sections satisfy the following objectives:

(1) Section II presents the results of the pilot application of the RAMSS to an on-going AFOTEC program. The program selected by AFOTEC for this application was the JTIDS Class 2 Terminal.

(2) Section III contains a discussion of the refinements to the RAMSS as a result of lessons learned during the pilot application. This discussion is quite technical in nature, involving statistical analysis techniques.

(3) Section IV contains a summary of the conclusions and recommendations of this study and pilot application.

(4) Appendix A contains a set of briefing materials which will be useful to AFOTEC in presenting or introducing the RAMSS to others.

(5) Appendix B is an Evaluator's Guide which can serve as stand-alone reference material for those who are applying the RAMSS to future programs. This material will guide the evaluator through the necessary steps of RAMSS and discuss the integration of the data into a report which concludes the entire process.

1.4 REFERENCES.

The following documents are referenced by this report:

(1) 1.4.1 "Software Supportability Risk Assessment: Pilot Application," Subtask Statement 412 for AFOTEC Contract F29601-85-C-0058, AFOTEC, Kirtland AFB, NM, October 1985

(2) 1.4.2 Hoessel, W., W. Huebner, D. Peercy, G. Richardson, "Software Supportability Risk Assessment in OT&E: Literature Review, Current Research Review, and Data Base Assemblage," BDM/A-84-0322-TR (Final), September 1984

(3) 1.4.3 Huebner, W., D. Peercy, G. Richardson, "Software Supportability Risk Assessment in OT&E: An Evaluation of Risk Methodologies," BDM/A-84-0496-TR (Final), August 1984

(4) 1.4.4 Huebner W., D. Peercy, G. Richardson, "Software Supportability Risk Assessment in OT&E: Measures for a Risk Assessment Model," BDM/A-84-0565-TR (Final), September 1984

(5) 1.4.5 Peercy, D., W. Huebner, M. Estill, J. Wu, "Software Supportability Risk Assessment in OT&E: Historical Baselines for Risk Profiles," BDM/A-85-0510-TR (Vols I and II), October 1985

(6) 1.4.6 Peercy, D., M. Estill, W. Huebner, K. Shaw, J. Wu, "Risk Assessment Methodology for Software Supportability (RAMSS) User's Handbook," BDM/A-85-1270-TR, April 1986

(7) 1.4.7 Peercy, D., W. Huebner, "Risk Assessment Methodology for Software Supportability (RAMSS): Guidelines for Adapting Software Supportability Evaluations," BDM/ABQ-86-0090-TR, April 1986

(8) 1.4.8 AFOTECP 800-2 Volumes I through V Software OT&E Guidelines. (Volume V is no longer being published.)

(9) 1.4.9 dBase III User Manual, Ashton Tate, Culver City, CA, 1984

(10) 1.4.10 BMDPC: User's Guide to BMDP on the IBM PC, BMDP Statistical Software, Inc., Los Angeles, CA, (no date)

(11) 1.4.16 Fisk, F., and W. Murch, "A Proposal for Computer Resources Risk Assessment During Operational Test and Evaluation," AFOTEC Draft Report, October 3, 1983.

1.5 TERMS AND ABBREVIATIONS.

AF	Air Force
AFB	Air Force Base
AFOTEC	Air Force Operational Test and Evaluation Center
ALC	Air Logistics Center
ASSET	AFOTEC Software Support Evaluation Tool
BMDP	BMDP Statistical Software (NOTE: BMDP is a name, not an acronym.)
CDRL	Contract Data Requirements List
C-E	Communications-Electronics
COMMANDS	Communication and Navigation Dynamic Simulator
CPIN	Computer Program Identification Number
CRISP	Computer Resources Integrated Support Plan
CRLCMP	Computer Resources Life Cycle Management Plan
CRMP	Computer Resources Management Plan

CRWG	Computer Resources Working Group
CSCI	Computer Software Configuration Item
CTP	Cipher Text Processor
DoD	Department of Defense
DSE	Deputy for Software Evaluation
ECS	Embedded Computer System
EPROM	Erasable, Programmable Read-Only Memory
ESD	Electronic Systems Division
FCA	Functional Configuration Audit
FQT	Formal Qualification Test
HQ-TAC	Headquarters, Tactical Air Command
ICPCP	Indicator Control Panel Control Program
IOT&E	Initial Operational Test and Evaluation
ISF	Integrated Support Facility
MCE	Modular Central Equipment
NICP	Network Interface Control Program
O/S CMP	Operational/Support Configuration Management Procedures
OT&E	Operational Test and Evaluation
PCA	Physical Configuration Audit
PMRT	Program Management Responsibility Transfer
PQT	Preliminary Qualification Test
PTP	Plain Text Processor
QA	Quality Assurance
QAP	Questionnaire Analysis Program
RA	Risk Assessment
RAMSS	Risk Assessment Methodology for Software Supportability
RFP	Request For Proposal
S/W	Software
SICP	Subscriber Interface Control Program
SLCP	Software Life Cycle Process
SPM	Software Product Maintainability
SS	Software Supportability
SSR	Software Support Resources
STM	Software Test Manager

TATP	Terminal Acceptance Test Program
TEMP	Test and Evaluation Master Plan
TPO	Test Plan Outline
USBE	User/Supporter Baseline Estimate
WR-ALC	Warner Robins Air Logistics Center

II.
Pilot Evaluation

SECTION II

JTIDS CLASS 2 TERMINAL PILOT EVALUATION

2.1 INTRODUCTION.

a. This section presents a brief background of the software supportability evaluation, the evolution of the User/Supporter Baseline Estimate, the evaluation procedure, results of the evaluation, analysis of the results and some lessons learned for future application of the RAMSS.

b. Part of the pilot study task to apply the Risk Assessment Methodology for Software Supportability (RAMSS) was to develop the details of the proposed Software Life Cycle Process (SLCP), evaluate the SLCP, and report the lessons learned from the evaluation effort. The refined procedures for the SLCP evaluation, evaluation questions, source of evaluation question requirement, and guidelines for the evaluation response are found in the Software Supportability Risk Assessment Evaluation Adaption Guidelines (reference 1.4.7).

2.2 BACKGROUND.

a. The was the system selected by AFOTEC for a pilot study on applying the Risk Assessment Methodology for Software Supportability (RAMSS). This methodology integrates evaluation data on the software product, software support resources, and software life cycle process to derive the risk to the Air Force of supporting the software. The software product and software support resources evaluation methodologies already are in use by AFOTEC (see reference 1.4.8). Details of the software life cycle process (SLCP) evaluation methodology were derived as part of the current support task SS 412 (see reference 1.4.7).

b. T[redacted] software supportability evaluations for IOT[redacted] were conducted at Eglin AFB during the period January 6-17, 1986. Prior to these evaluations, plans and preparations were performed by AFOTEC and the BDM Corporation SS 412 personnel. In addition, several meetings were held with AFOTEC, WR-ALC, and HQ-TAC personnel to establish an initial User/Supporter Baseline Estimate on the software support concept and change profile.

c. The software product evaluation was conducted January 6-17 using four evaluators from WR-ALC , and one evaluator from TAWC/ . The AFOTEC Software Test Manager (STM) and Deputy for Software Evaluation (DSE) provided appropriate calibration assistance. The software support resources evaluation was conducted on January 9 and 10 using the five evaluators of the software product as well as one BDM representative. The software life cycle process evaluation was conducted by the BDM representative in parallel with the software support resources evaluation.

2.3 EVOLUTION OF THE USER/SUPPORTER BASELINE ESTIMATE.

a. One important part of the pilot study was to determine the effort and procedures required to obtain a user/supporter baseline estimate of the software support concept and change profile over the first few block releases. Another important part was to determine the impact of using (and not using) such an estimate during the software supportability evaluation.

b. The user/supporter baseline estimate is simply an estimation of the support resources and software change activity expected for a given software system for one or more block releases during post deployment software support. This "estimate" or "concept" is derived by reviewing historical software maintenance data, available acquisition planning information in documents such as the CRISP or O/S CMP, the current software system status (e.g., maturity, current development and support change activity), and the views of the using and supporting command personnel. The process may iterate until a reasonable consensus or compromise is derived. This user/supporter baseline estimate then becomes a baseline against which software supportability evaluation measures can be derived and the software's supportability risk computed.

c. Evolution of the user/supporter baseline estimate is described in the next several paragraphs. Use of the estimate is described in later sections and the conclusions are summarized in section 2.7.

2.3.1 The User/Supporter Baseline Estimate Evolution Process. The user/supporter baseline estimate evolved as a series of interface discussions among the using command (HQ-TAC) representatives, the supporting command (WR-ALC) representatives, and AFOTEC/BDM representatives. The sequence of events is listed below:

(1) Visit with HQ-TAC by AFOTEC/BDM (November 14, 1985)

(2) Visit with WR-ALC by AFOTEC/BDM (November 15, 1985)

(3) First draft generated by BDM (November 22, 1985)

(4) Review of first draft by HQ-TAC and WR-ALC personnel (December 5, 1985)

(5) Compromise for second draft distributed to HQ-TAC/WR-ALC (December 10, 1985)

(6) Final revision of estimate generated during evaluation (January 9, 1986).

2.3.2 **First Step: Visit with Using Command (HQ-TAC).**

a. Using command personnel from HQ-TAC/ provided valuable insight into the derivation of support requirements and the need for those requirements from the user perspective. Several issues from the current CRISP were discussed and the concept of the user/supporter baseline estimate was presented.

b. From the user perspective, a new version of software will be released to the user approximately every 6 months. The user thus will have one version in the field and at least one version undergoing system field test prior to being released to the field. The 6-month "cycle" is based upon the Philosophy and the current 6-month release cycle for the Equipment ( ) which will eventually replace the system. There would be some overlap in the 6-month release cycle workload for support personnel (see the WR-ALC discussion for more detailed information on the release cycle).

c. The personnel allocation in the draft CRISP, Volume III, January 1985, included a general manning level of 16 persons and an additional 5 persons dedicated to specific software support. The CRISP was not specific as to all the software which will be supported, nor the distribution of the 16 and 5 persons across the software systems. The using command personnel could not offer any further clarification other than to indicate five persons were not enough. The latest draft CRISP (November 1985) has even removed specific personnel requirements such as the number and function.

d. Potential areas which will increase the software supportability risk of the include:

(1) Emergency changes

(2) Operational Interfaces

(3) Incomplete Development (23 of 88 messages have not been defined)

(4) Six-Month Release Cycle (must be achieved to meet releases)

(5) Lack of personnel dedicated to the soft-ware support.

e. The using command personnel suggested that AFOTEC put a form of the user/supporter baseline estimate in the Test Plan Outline (TPO) and evolve the user/supporter baseline estimate information along with the TPO. They also suggested there would need to be clear guidance (e.g, authority) in order to require such an estimate and require the maintenance data collection necessary to keep the historical data base up to date. Further discussions with AFOTEC have indicated that the CRISP or the Computer Resources Life Cycle Management Plan (CRLCMP) may be better documents to contain the user/supporter baseline estimate.

2.3.3 <u>Second Step: Visit with Supporting Command (WR-ALC)</u>.

a. Supporting command personnel from WR-ALC/ and WR-ALC/ provided valuable insight into the current support of the and the requirements for software support. The currently estimated block release cycle, support personnel levels, and software systems to be supported as described in the draft CRISP, Volume III, January 1985, were discussed. Guidance useful for deriving the necessary user/supporter baseline estimate was obtained.

b. The software systems to be supported as part of the support include:

(1) Network Interface Control Program (NICP)

(2) Subscriber Interface Control Program (SICP)

(3) Indicator Control Panel Computer Program (ICPCP)

(4) Plain Text Processor (PTP)

(5) Cipher Text Processor (CTP)

(6) Integrated Support Facility (ISF) Support Software.

The ICPCP software development is the responsibility of the Army. All other software systems are the development responsibility of the Air Force.

c. The support release cycle will actually be about 9 months with a 3-month overlay for analysis of release change content. This will result in a new version being released to the user approximately every 6 months. The full 9 months is for engineering support and does not include technical orders, prom burning, and final field distribution and test.

d. The actual personnel being used for the soft- ware support includes nine persons for analysis, design/implementation, and test, and six persons for ISF support. The personnel have skill/experience levels ranging from 2 to 8 years and are evenly distributed across the RAMSS personnel skill levels of 2, 3, 4. Note that the RAMSS uses five personnel skill levels, from 1 (lowest, meaning entry level personnel) to 5 (highest, meaning the most skilled and experienced personnel). It is expected that three of the nine (plus one other person) will have similar support functions and two of the six persons will have similar ISF support functions.

e. is allocated a total of approximately 43 support personnel for all functions (including software support). The formula for determining resource allocations was thought to be in a Joint Logistic Decision Tree Analysis AFLC manual. Support command personnel agreed the current CRISP allocation of 16 general support personnel and 5 support personnel was not sufficient. At least double the allocation of five is required.

f. It is perceived that there is lower support risk from having a single developer (rather than multiple developers as for the). Also, the Integrated Support Facility (ISF) is considered to be low risk. Experience of support personnel on the indicates it takes <u>much more time</u> than anticipated for customizing contractor procedures to the support facility, and repeating contractor qualification tests following <u>initial</u> software support changes. The learning curve progressed slower than expected for these activities.

g. Support command personnel discussed several areas of concern which would increase the software supportability risk:

(1) Plain Text Processor (PTP) - this firmware (card A1 of the Digital Data Processor) is not documented and could be subject to change

(2) Cipher Text Processor (CTP) - this firmware (card A6 of the Digital Data Processor) is not documented and could be subject to change

(3) Poor documentation of the operational software

(4) Security - code is classified as confidential and will cause delays in support response due to a more secure support environment and security procedures

(5) Unique Test Boxes - some one-of-a-kind test support boxes developed by the contractor may become an availability/upgradeability problem during support

(6) Test Tricks - procedures for some known test tricks (e.g., EPROM write patch, 6-second reset switch) may not be documented.

h. On the basis of discussions with the support personnel, an initial allocation of personnel was defined for input to the first draft of the user/supporter baseline estimate. The initial allocation is shown in table 2-1. Support personnel also provided an understanding of the anticipated block release schedule (see figure 2-1).

Table 2-1.

Initial Support Personnel Allocation

FUNCTION	NUMBER OF PERSONNEL	NUMBER IN EACH SKILL LEVEL (1..5) (1 = LOWEST, 5 = HIGHEST)	% DEDICATED CLASS 2	% DEDICATED RELEASE 1
General Support	16	(2,3,4,4,3)	10%	83%
Dedicated Support (NICP)	10	(1,2,3,2,2)	30%	83%
(SICP)	10	(1,2,3,2,2)	30%	83%
(ICPCP)	10	(1,2,3,2,2)	20%	83%
(Other)	10	(1,2,3,2,2)	15%	83%
External System Support (5%)	10	(1,2,3,2,2)	0%	0%

86-0360-TR-W-01

i. An estimate of change requests for the various software systems for the first few block releases was not obtained directly. However, some estimates on maximums for different categories were done (see table 2-2). These estimates along with the general distribution of changes across the categories were used to obtain an initial draft for the user/supporter baseline estimate. The "other" category includes ISF, PTP, and CTP software/firmware support.

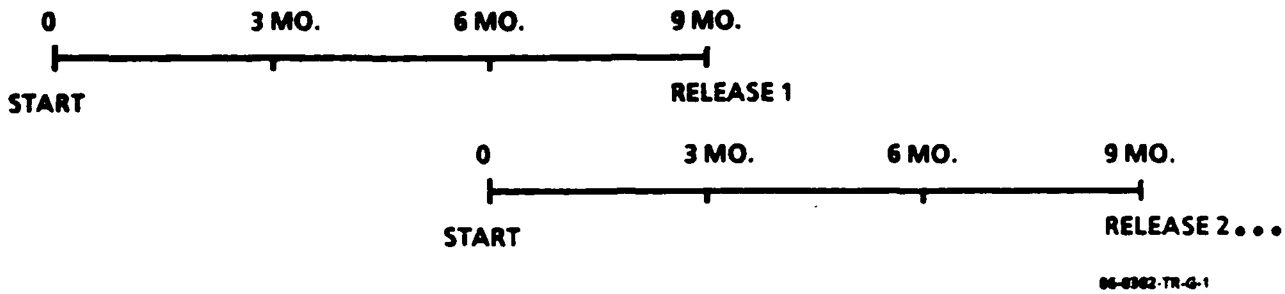


Figure 2-1. Block Release Schedule

Table 2-2.

Estimates of Software Change Requests

			TYPE			COMPLEXITY			PRIORITY		
SW SYSTEM	BLOCK RELEASE	TOTAL# CHANGES	#C	#H	#V	#H	#M	#L	#E	#U	#N
NICP	1	9	9	0	0	0	0	9	0	0	9
	2	12	10	2	0	1	4	7	0	1	11
	3	15	12	2	1	1	5	9	1	1	13
SICP	1	9	9	0	0	0	0	9	0	0	9
	2	12	10	2	0	1	4	7	0	1	11
	3	15	12	2	1	1	5	9	1	1	13
ICPCP	1	5	5	0	0	0	0	5	0	0	5
	2	6	5	1	0	0	2	4	0	1	5
	3	8	5	2	1	1	2	5	1	1	6
Other	1	3	3	0	0	0	0	3	0	0	3
	2	5	4	1	0	0	1	4	0	1	4
	3	8	5	2	1	1	2	5	1	1	6
TOTALS (all Class 2 S/W)	1	26	26	0	0	0	0	26	0	0	26
	2	35	29	6	0	2	11	22	0	4	31
	3	46	34	8	4	4	14	28	4	4	38

TYPE:	COMPLEXITY:	PRIORITY:
C = CORRECTION	H = HIGH	E = EMERGENCY
H = ENHANCEMENT	M = MEDIUM	U = URGENT
V = CONVERSION	L = LOW	N = NORMAL

86-0360-TR-W-014

2.3.4 Third Step: Generate First Draft of Estimate.

a. The first draft of the user/supporter baseline estimate was generated from information gathered in steps 1 and 2 as well as the maintenance release data base (RAMSS User's Handbook, see reference 1.4.6). Ordinarily, this estimate could have been derived using the RAMSS automated support tool, but the tool was not available when the first draft was generated. A first draft of the estimate could also have been derived from the maintenance release data base prior to the visits and then used during the visits as a starting point for discussion.

b. Another interesting feature which was not available during the generation of the estimate is the estimation of the required person months per change from a regression equation using independent parameters such as average skill level, type/complexity/priority of baseline profile change requests, and the type of software system (e.g. Communications-Electronics). See section 3.5 for more details on this feature. For purposes of future comparison, this feature is integrated on each report of the baseline estimate which displays estimated risk (the report is a product of the RAMSS automated support tool).

c. The first draft user/supporter baseline estimate results are shown in figure 2-2, which is a report generated by the RAMSS automated support tool. The evolution of the release schedule and support staff from the visits with HQ-TAC and WR-ALC personnel is apparent. The baseline support profile summarizes the changes across all software systems which are to be supported. The available person months per change is simply the full time equivalent person months (computed by using the percent dedicated and release overlap information) divided by the total number of changes. The estimated person months per change is derived from a regression equation and represents a more realistic estimate of the optimum required resources based upon

EVALUATION REPORT A1: USER /SUPPORTER BASELINE CONCEPT

```
******************************************************************
*                        SYSTEM PROFILE                          *
*                                                                *
* SWSYSID   :                        SWTYPE    : C-E             *
*                                                                *
* SYSTEM    :                        SUPPORTER: WR-ALC           *
*                                                                *
* SWSYSTEM :                         USER      : HQ-TAC          *
******************************************************************
```

SUPPORT CONCEPT

Release Schedule: 9.00 Month block release cycle with 3.00 Month overlap

Support Staff : 16 Persons, 10% Dedicated, Avg.skill level 3.00

10 Persons, 95% Dedicated, Avg.skill level 3.00

BASELINE SUPPORT PROFILE

BLOCK	TOTAL # CHANGES	TYPE(C,H,V)	COMPLEXITY(H,M,L)	PRIORITY(E,U,N)
1	26	(26, 0, 0)	(0, 0, 26)	(0, 0, 26)
2	35	(29, 6, 0)	(2, 11, 22)	(0, 4, 31)
3	46	(34, 8, 4)	(4, 14, 28)	(4, 4, 38)

ESTIMATED SOFTWARE SUPPORTABILITY RISK

BLOCK	AVAILABLE PERSON MONTHS PER CHANGE	ESTIMATED PERSON MONTHS PER CHANGE	ESTIMATED RISK
1	3.20	1.98	0.31
2	1.90	2.48	0.61
3	1.45	2.89	0.77

Figure 2-2. First Draft of Pilot Study User/Supporter Baseline Estimate

the skill of personnel and the workload complexity of the task. There is risk for any level of person months per change, and this estimated risk is shown for each of the first three block releases. The estimated risk (see Glossary for precise definition) represents the likelihood that resources (personnel and schedule) will not be adequate to meet the particular baseline change profile block workload. The estimated risk is computed by integrating over a normal distribution with mean (the estimated person months per change) and standard deviation (standard estimate of error from the regression equation) from the available person months per change to infinity.

d. Optimum utilization of resources occurs when the available and estimated persons months per change are the same. This does not mean the estimated supportability risk is minimal, or even low. There is a tradeoff between lowering risk by increasing resources and having a more optimal (for the estimated workload) level of resources.

2.3.5 Fourth Step: Review of First Draft by HQ-TAC and WR-ALC Personnel.

a. The Computer Resources Working Group (CRWG) for the met on December 4-5, 1985, at AFOTEC. It was intended that BDM personnel would discuss the draft user/supporter baseline estimate for the software risk assessment method shortly following the CRWG, since both the using and supporting commands had planned to attend the meeting. The supporting command representatives from WR-ALC were present, however a last minute schedule change for the using command representatives in HQ-TAC at Langley AFB prevented their attendance. Rather than miss the opportunity to establish an update to the user/supporter baseline estimate, a conference telephone call was performed with the required HQ-TAC and WR-ALC participants, beginning at 11:30 AM on December 5, 1985, and lasting for about 1 hour.

b. Prior to the telephone conversation, inputs on the first draft were made by the WR-ALC personnel. These inputs were discussed with the HQ-TAC personnel during the telephone conversation.

c. The purpose of the telephone conversation was to obtain agreement on three basic issues:

(1) The number of support personnel required to maintain the software

(2) An understanding of, and agreement on, the software block release cycle for

(3) The total number of anticipated changes, grouped by type, complexity, and priority, for the first three block releases.

d. Since both the using and supporting command representatives agreed with the first draft estimate on items 1 and 2 above, the remainder of this section will discuss the evolution of item 3 and the proposed baseline support profile for this data.

e. As shown in figure 2-3, the total changes predicted by the first draft estimate increased with each block release. This data was not meant to predict that the total changes would increase with all succeeding block releases, but was derived from the knowledge that most software maintenance projects have increased change activity for a period of time after operational release. After reaching a peak, the change activity normally decreases with time. Change activity data in the various categories (type, complexity, priority) were chosen as representative of the historical data previously collected for systems like . Change activity data was based upon planned changes to all software, to include:

(1) Network Interface Control program (NICP)

(2) Subscriber Interface Control Program (SICP)

(3) Indicator Control Panel Computer Program (ICPCP)

(4) Plain Text processor (PTP)

(5) Cipher Text processor (CTP)

(6) Integrated Support Facility (ISF) Support Software.

		TOTAL CHANGES	TYPE			COMPLEXITY			PRIORITY		
			C	H	V	H	M	L	E	U	N
BLOCK 1	DRAFT	26	26	0	0	0	0	26	0	0	26
	SUPPORTER	11	11	0	0	0	0	11	0	0	11
	USER	26	26	0	0	0	0	26	0	0	26
	COMPROMISE	15	15	0	0	0	0	15	0	0	15
BLOCK 2	DRAFT	35	29	6	0	2	11	22	0	4	31
	SUPPORTER	12	9	3	0	0	1	11	0	0	12
	USER	35	29	6	0	2	11	22	0	4	31
	COMPROMISE	20	15	5	0	1	4	15	1	4	15
BLOCK 3	DRAFT	46	34	8	4	4	14	28	4	4	38
	SUPPORTER	9	6	3	0	1	1	7	0	0	9
	USER	46	34	8	4	4	14	28	4	4	38
	COMPROMISE	20	13	7	0	1	6	13	1	6	13

TYPE:	COMPLEXITY:	PRIORITY:
C = CORRECTION	H = HIGH	E = EMERGENCY
H = ENHANCEMENT	M = MEDIUM	U = URGENT
V = CONVERSION	L = LOW	N = NORMAL

86-0360-TR-W-02

Figure 2-3. Pilot Study Development of User/Supporter Baseline Change Profile

f. As indicated in figure 2-3, the supporting command representatives predicted much less maintenance activity than the draft. Part of this prediction was based on the experience gained supporting the , as well as an expectation that the PTP, CTP, and ISF support software will not be maintained by WR-ALC personnel. According to remarks made at the CRWG, changes made to the PTP and CTP software will probably be contracted out since there is no documentation for this software unless ESD funds a documentation effort. Other significant predictions by the supporting command representatives included: (1) a peak of 12 changes will occur in block 2, decreasing to 9 changes in block 3; (2) there will be no conversion activity; (3) there will be few medium or high complexity changes; and (4) there will be no urgent or emergency changes.

g. The telephone conversation with the using command at HQ-TAC revealed a much different opinion. The using command representatives commented that, based on their experience, the original draft baseline support profile seemed more realistic. In particular, they contended that the supporting command predictions: (1) were much too low in total number of expected changes; (2) did not anticipate the potential requirement for urgent or emergency changes; and (3) did not account for the historical information gathered on similar systems regarding change activity distribution. The using command did not give specific predictions in each category, but indicated that the supporting command should seriously consider the using command experience and opinion in establishing the baseline. Specific numbers were not agreed upon at the end of the conversation.

h. Immediately following the telephone conversation, discussion was continued with the supporting command representatives regarding the baseline support profile. The results of the discussion are indicated under "COMPROMISE" in figure 2-3. In general, the supporting command agreed to increase the predicted number of changes and to use historical data as guidelines, although not to the extent of the first draft recommendations.

i. Each of the change profiles discussed above have associated estimated risks which correspond to the projected software support requirements. A summary of these risks is shown in figure 2-4.

		AVAILABLE PERSON TIME PER CHANGE (mo) (PMPC)	ESTIMATED PMPC	ESTIMATED RISK
BLOCK 1	DRAFT	3.20	1.98	0.31
	SUPPORTER	7.57	1.98	0.08
	USER	3.20	1.98	0.31
	COMPROMISE	5.55	1.98	0.14
BLOCK 2	DRAFT	1.90	2.48	0.61
	SUPPORTER	5.55	1.96	0.14
	USER	1.90	2.48	0.61
	COMPROMISE	3.33	3.11	0.47
BLOCK 3	DRAFT	1.45	2.89	0.77
	SUPPORTER	7.40	2.41	0.12
	USER	1.45	2.89	0.77
	COMPROMISE	3.33	3.57	0.53

86-0360-TR-W-03

Figure 2-4. Pilot Study Risk by Block Release for User/Supporter Baseline Change Profile

j. These risks represent the potential for unwanted, negative consequences, such as not meeting the desired block release schedule. For example, in Block 1 the estimated risk is .14 (compromise), meaning that there is a 14 percent chance that the projected software support resources will not be able to support the predicted support activity. The estimated risks are higher for the successive block releases, as shown.

2.3.6 Fifth Step: Compromise for Second Draft. The user/supporter baseline estimate takes into account the opinions, experience, and knowledge of both the using and supporting commands. The recommended baseline estimate (the second draft) is illustrated in figure 2-5. Comments similar to those applied to the first draft in section 2.3.4 are applicable. This estimate was used as an initial input to the software supportability evaluation process.

EVALUATION REPORT A1: USER /SUPPORTER BASELINE CONCEPT

```
****************************************************************
*                         SYSTEM PROFILE                       *
*                                                              *
* SWSYSID   :                        SWTYPE    : C-E           *
*                                                              *
* SYSTEM    :                        SUPPORTER: WR-ALC         *
*                                                              *
* SWSYSTEM  :                        USER      : HQ-TAC        *
****************************************************************
```

SUPPORT CONCEPT

Release Schedule: 9.00 Month block release cycle with 3.00 Month overlap

Support Staff : 16 Persons, 10% Dedicated, Avg.skill level 3.00

10 Persons, 95% Dedicated, Avg.skill level 3.00

BASELINE SUPPORT PROFILE

BLOCK	TOTAL # CHANGES	TYPE(C,H,V)	COMPLEXITY(H,M,L)	PRIORITY(E,U,N)
1	15	(15, 0, 0)	(0, 0, 15)	(0, 0, 15)
2	20	(15, 5, 0)	(1, 4, 15)	(1, 4, 15)
3	20	(13, 7, 0)	(1, 6, 13)	(1, 6, 13)

ESTIMATED SOFTWARE SUPPORTABILITY RISK

BLOCK	AVAILABLE PERSON MONTHS PER CHANGE	ESTIMATED PERSON MONTHS PER CHANGE	ESTIMATED RISK
1	5.55	1.98	0.14
2	3.33	3.11	0.47
3	3.33	3.57	0.53

Figure 2-5. Second Draft of Pilot Study User/Supporter Baseline Estimate

2.3.7 **Sixth Step: Final Revision and Use of the User/Supporter Baseline Estimate During the Software Supportability Evaluation.** The second draft of the user/supporter baseline estimate was presented to the evaluators. The evaluators included the same WR-ALC supporting command personnel who had commented on earlier draft versions. A more thorough analysis of personnel requirements had been conducted by the supporting command personnel. The new allocation of personnel is shown below:

(1) 2 persons: 10 percent dedicated (supervisors)

(2) 13 persons: 20 percent dedicated (general support)

(3) 9 persons: 90 percent dedicated (direct support).

This allocation was slightly different than the previously established allocation in the second draft (10.9 full-time equivalents versus 11.1 full-time equivalents). The final draft of the user/supporter baseline estimate for the is illustrated in figure 2-6.

2.4 EVALUATION PROCEDURE.

a. The evaluation procedures for the software product and software support resources were essentially as directed by the AFOTECP 800-2 guidelines (see reference 1.4.8). The primary change to those procedures was the discussion of the user/supporter baseline estimate as it had finally evolved (see section 2.3). This discussion helped orient the evaluators to the estimated personnel resource requirements and change profile workload. This orientation was the only noticeable use of the baseline estimate during the evaluation process, but was considered very helpful to the evaluation participants. Discussion of individual evaluation questions did not involve reference to the baseline estimate.

EVALUATION REPORT A1: USER /SUPPORTER BASELINE CONCEPT

```
*****************************************************************
*                        SYSTEM PROFILE                          *
*                                                                *
* SWSYSID   :                       SWTYPE    : C-E              *
*                                                                *
* SYSTEM    :                       SUPPORTER: WR-ALC            *
*                                                                *
* SWSYSTEM  :                       USER      : HQ-TAC           *
*****************************************************************
```

SUPPORT CONCEPT

Release Schedule: 9.00 Month block release cycle with 3.00 Month overlap

Support Staff : 15 Persons, 19% Dedicated, Avg.skill level 3.00

9 Persons, 90% Dedicated, Avg.skill level 3.00

BASELINE SUPPORT PROFILE

BLOCK	TOTAL # CHANGES	TYPE(C,H,V)	COMPLEXITY(H,M,L)	PRIORITY(E,U,N)
1	15	(15, 0, 0)	(0, 0, 15)	(0, 0, 15)
2	20	(15, 5, 0)	(1, 4, 15)	(1, 4, 15)
3	20	(13, 7, 0)	(1, 6, 13)	(1, 6, 13)

ESTIMATED SOFTWARE SUPPORTABILITY RISK

BLOCK	AVAILABLE PERSON MONTHS PER CHANGE	ESTIMATED PERSON MONTHS PER CHANGE	ESTIMATED RISK
1	5.47	1.98	0.14
2	3.29	3.11	0.48
3	3.29	3.57	0.54

Figure 2-6. Final Draft of Pilot Study User/Supporter Baseline Estimate

b. The software product evaluation was independently conducted in parallel with the other software supportability evaluations. In preparation for the software support resources (SSR) and software life cycle process (SLCP) evaluations, AFOTEC and BDM participants met for 2 hours on January 8 in order to review the SSR evaluation questionnaires, discuss the evaluation rating criteria, and set a schedule agenda for the SSR and SLCP evaluations. It was decided that a good approach would be to introduce the evaluation group to the agenda and schedule, briefly discuss the three areas of SSR evaluation, conduct the SSR evaluation by area (completing responses to evaluation questions after each discussion), and then discuss the SLCP evaluation elements. Questionnaires were completed, copies made for evaluators, and introductory slides were prepared.

c. The SSR evaluation was conducted according to the prepared agenda, although the original time schedule was lengthened. The SSR evaluation was conducted at a level slightly below the RAMSS required level so the results have been accumulated to arrive at the values required in the three areas of personnel, support systems, and physical facilities. The evaluators completed responses to questions after sometimes lengthy discussions. Some questions were combined to provide proper balance.

d. Each question was answered relative to the adequacy of the subject addressed. Characteristics of "adequacy", i.e., "what does adequate mean - how does one judge adequacy", were also discussed. For example, characteristics such as planning, funding level, performance, documentation, quantity, and so forth were used as appropriate.

e. The discussions related to the SSR evaluation were very helpful in understanding the plans for and actual status of the SSR. Generally, the CRISP and O/S CMP are very poor in describing this information. There seems to be much more capability than is being reflected in these "planning" documents, and in effect the documents are not being used for planning at all. A more complete description of the software product/software support resources evaluation process and results will be produced by AFOTEC OT&E personnel. Results necessary for input to the RAMSS are described in section 2.5.

f. An overview of the two SLCP major factors, project management and configuration management, was presented by the BDM participant. These factors were discussed as processes conducted by the three activities: procurement, development contractor, and operation support. Many deficiencies, positive attributes and rationale were discussed. A subjective consensus rating was obtained from the discussion participants in the various subelements across the activities. These ratings, the discussion, and review of documents have been integrated to arrive at the required SLCP evaluation for input to the RAMSS pilot study. These results are described in section 2.5.

g. It was very apparent from the discussion that several more sessions could have been spent without covering all the issues. Some of the problems included short procurement schedule, lack of adequate procurement configuration management identification, initial lack of coordination between contractor and procurement activity, procurement organizational structure (lack of continuity), contractor management of subcontractor, and so forth. The contractor test strategy was considered to be very complex, but thorough. Operational/Support configuration management was considered poor.

h. Although this discussion was very valuable for the SLCP evaluation, the recommended procedure is for the AFOTEC Software Test Manager and Deputy for Software Evaluation to accumulate the required SLCP data over a longer period of time from program reviews, resource working groups, procurement meetings, and so forth. There is simply too much information which needs to be integrated across all life cycle phases and all responsible activities. Specific guidelines are presented in the Software Life Cycle Process Evaluator's Guide in reference 1.4.7.

i. All necessary evaluation data were collected for the RAMSS pilot study, and the required software supportability evaluation was completed satisfactorily. More time could have been used for both the SSR and SLCP discussions.

2.5 EVALUATION RESULTS.

a. The evaluation results required for input to the RAMSS pilot study include values from the three software supportability level 1 criteria (see figure 2-7): software life cycle process, software product maintainability, and software support resources. The evaluation results for the software life cycle process were derived by BDM. The evaluation results for the software product maintainability and the software support resources were derived by AFOTEC during the planned IOT&E of the software.

b. The software supportability evaluation results (on a scale of 1.0 (low) to 6.0 (high)) at the lowest level 3 required for input to the RAMSS are shown in figure 2-8, along with the accumulated average values at the level 2 and level 1 of the evaluation hierarchy. The overall supportability score and evaluated risk are also shown. The overall supportability score is the unweighted average of the level 1 results. The evaluated risk is on a scale of 0.0 (no risk) to 1.0 (absolute certainty). See Glossary for more information on evaluated risk.

2.5.1 Software Life Cycle Process Evaluation Results. Detailed questions for this evaluation, as described in the Software Life Cycle Process Evaluator's Guide (see reference 1.4.7), were only informally used since all questions were not available during the

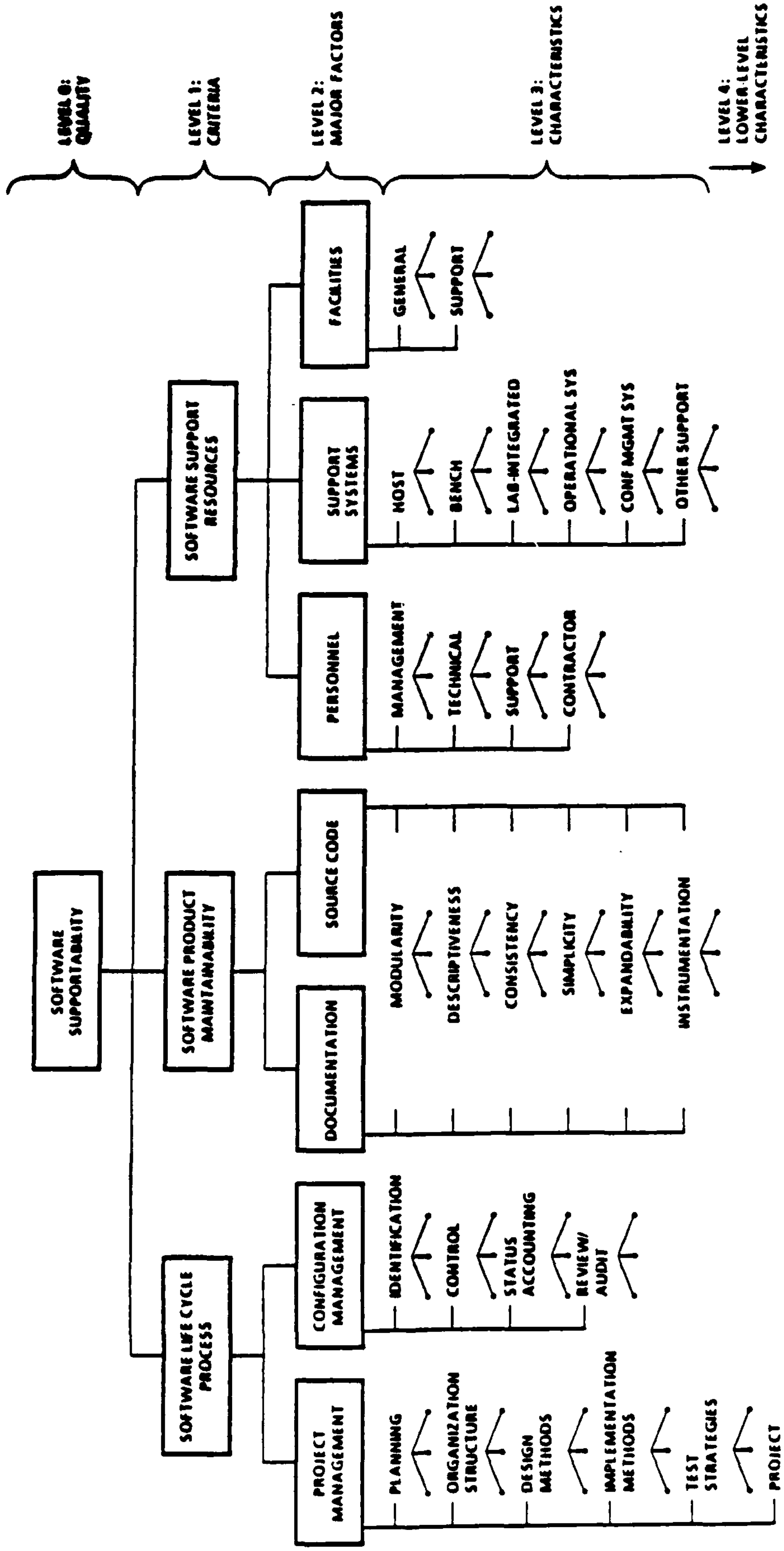


Figure 2-7. Elements of Software Supportability Evaluations

EVALUATION REPORT A2: TABLE OF EVALUATION SCORES

```
*****************************************************************
*                        SYSTEM PROFILE                         *
*                                                               *
* SWSYSID   :                        SWTYPE    : C-E            *
*                                                               *
* SYSTEM    :                        SUPPORTER: WR-ALC          *
*                                                               *
* SWSYSTEM  :                        USER      : HQ-TAC         *
*****************************************************************
```

I. LIFE CYCLE PROCESS	3.32	II. PRODUCT	4.15	III. SUPPORT RESOURCES	3.94
A. PROJECT MANAGEMENT	3.47	A. DOCUMENTATION	3.97	A. PERSONNEL	3.53
1. Planning	3.33	1. Modularity	4.70	1. Manager	3.75
2. Organizational Structure	3.33	2. Descriptiveness	3.50	2. Technical	3.38
3. Design Methods	4.00	3. Consistency	3.80	3. Support	3.17
4. Implementation Methods	3.50	4. Simplicity	3.90	4. Contractor	3.83
5. Test Strategies	3.67	5. Expandability	3.40		
6. Project Interface	3.00	6. Instrumentation	4.50	B. SUPPORT SYSTEMS	3.72
				1. Host	4.02
B. CONFIGURATION MANAGEMENT	3.17	B. SOURCE LISTINGS	4.32	2. Bench	4.07
1. Identification	3.33	1. Modularity	5.20	3. Laboratory	0.00
2. Configuration Control	3.50	2. Descriptiveness	4.20	4. Operational	4.11
3. Status Accounting	3.00	3. Consistency	3.50	5. CMS	2.92
4. Audit	2.83	4. Simplicity	5.00	6. Other	3.50
		5. Expandability	4.60		
		6. Instrumentation	3.40	C. FACILITIES	4.58
				1. General	4.50
				2. Support Systems	4.67

Computed Overall Score (Average of I, II, and III): 3.80

Software Supportability Confidence Assessment : 0.70

Software Supportability Evaluated Risk : 0.55

Figure 2-8. Pilot Study Software Supportability Evaluation Results evaluation and would ordinarily be answered over an extended period of time. Values shown in figure 2-8 were derived at the level 3 characteristics and accumulated to the higher levels in the usual fashion of averaging the associated immediate lower level values. No weighting is used in the RAMSS computations although the level 3 inputs could have been weighted prior to their input to RAMSS.

2.5.1.1 Software Project Management Evaluation Results.

a. The rationale behind the software project management evaluation results is briefly discussed in the following paragraphs.

b. The planning for the has been characterized by a very unrealistic procurement activity schedule, lack of TEMP and CRISP as well-detailed and implemented planning documents, and a non-existent O/S CMP. These plans and procedures are critical for all aspects of software supportability. In addition, lack of planning for computer resource acquisition of the integrated support system was evident since there were no initial plans to acquire software bench, and integrated operational test beds. This planning gap resulted in uncertainty about what development support systems were being acquired and, once determined, in poor documentation of these support systems. Contractor planning has been somewhat better than the procurement and support activities, but changes in project requirements as well as additional contract workload have caused continual redirection of the contractor resources. Major requirement changes included a bilingual capability and the packing of messages. After being determined, the software bench (COMMANDS) and the operational integrated test bed (TATP) station software were tasked as deliverables, but the hardware was not. The COMMANDS and TATP stations were defined as a required development resource, but the contractor obtained other contracts (e.g., the United Kingdom) for which these resources (systems and personnel) were to be used. The resource drain on the contractor resulted in the procuring activity requirement to purchase additional support resources. The procurement activity, contractor activity, and support activity were rated 3.0, 4.0, and 3.0 for an overall score of 3.33 for project planning.

c. The procurement activity organization structure has had numerous problems because of its lack of centralized control and personnel consistency throughout the project life cycle. Continual redirection results when lead personnel are changed or the authority for decision control changes or does not exist. The organization structure for procurement also suffered due to the interservice participation of the Army and Air Force. The Army was not involved in the CRWG and initially wanted all reference to the Army removed from the CRISP. After an attempt to write a separate Army Computer Resources Management Plan (CRMP), the Army again became a participant in the CRISP. Neither the CRISP nor CRWG (nor support resources in general) received proper priority within the procurement activity project structure with the Air Force as lead agency or the Army as the deputy agency. As for personnel consistency, since the beginning of full-scale development around 1981, WR-ALC support activity has had three complete group turnovers, program management has had three changes in leadership, configuration management office has had two leadership changes, the test branch and budget have each had three leadership changes, and the logistics branch has had two leadership changes. The contractor also had problems with staff attrition and internal organizational relationships with quality assurance and its subcontractor. The latter quality assurance interface is now much better. The organization structure for the support activity is getting better over time as more emphasis is put upon the post-deployment support. However, the CRISP and O/S CMP should reflect the support activity organization structure and these documents do a very poor job of documenting this structure. In fact, the latest version of the CRISP had removed the few specific requirements for support personnel, and the O/S CMP was just being written. The procurement activity, contractor activity, and support activity were rated 3.0, 4.0, and 3.0 respectively, for an overall score of 3.33 for project organization structure.

d. The design methods of the various activities were more difficult to evaluate without actually looking at the pertinent design documents and following the various design reviews. However, some observations are in order. First, the procurement activities original prototype design of the was thought to be good. Thus, early concept design must have been reasonably good. However, the full scale development procurement activity caused some perturbations in the design through changing requirements (bilingual capability and message packing). The initial system/segment specification is the procurement activity design document, and it must have been reasonably adequate even though the changing requirements have caused some schedule slippage. The contractor activity has good documented design methods as evidenced by the internal standards, design documents (e.g., Computer Program Development Plan), design reviews, test design, and use of structured design methods. The support activity has no evidence of internal standards manuals, intent to transition the contractor design methodology, or experience with modern design methods such as data flow, Yourdon Hierarchy Charts, object-oriented programming, or design standards/conventions. The procurement activity, contractor activity, and support activity were rated 4.0, 5.0, and 3.0 respectively, for an overall score of 4.0.

e. The implementation methods of the procurement activity were poor since no particular standards or requirements have been rigorously enforced. Contractor QA and internal standards are in place, but only after a struggle. The Mitre Corporation had been assigned the Government QA monitor role and the contractor had no specifically required QA reporting functions. Even now, however, the QA standards are not generally implemented. The inconsistencies between contractor standards as indicated in the Computer Program Development Plan and the resulting products such as the documentation and source code indicate that good design methods and techniques have not been implemented in the product forms. As examples, a naming convention for global and local variables was established but not rigorously followed; instrumentation standards were defined, but not followed. The contractor now has at least one person full time and others part-time performing the QA function for the

. The contractor has had very good configuration management control. The contractor unit test, integration, and system implementation process has been complex but thorough. At least a year slippage in schedule is due to this thoroughness, and perhaps the benefit will be seen in the post-deployment support. Support activity implementation methods have not been carefully defined. Standards and conventions are not defined to the level of detail necessary. The procurement activity, contractor activity, and support activity were rated 3.0, 4.5, and 3.0, respectively, for an overall score of 3.5 for implementation methods.

f. The test strategies for the procurement activity have been very poor from the initial lack of insight and planning for acquisition of adequate development test beds to the current lack of direction as to where the purchased test beds will actually be used during post-deployment support. The current DT&E activity is continually fighting problems with hardware and software in order to stay up long enough to conduct tests. It does not appear that there is a coordinated strategy between DT&E and OT&E agencies to share test data and strategies to optimize resources and effectiveness of the T&E process. It does not appear that there is adequate joint service coordination (Army and Air Force) of the test process. On the bright side, the contractor has apparently done a very thorough job of developing a phased test plan, test description, test procedures, and configuration control of the tests as a test suite to be transitioned to the supporting agency. The FQTs and PQTs executed by the contractor have been excellent. However, the thoroughness of the test strategy has caused a significant schedule slippage in the project (perhaps a year). Some criticism of the lack of priority in testing critical items was noted. Since the test process will be transitioned to the support activity, the benefit might be seen in the post-deployment support. However, the support activity will require more qualified personnel to understand and configure control such a complex test strategy. The supporting activity needs to make specific arrangements for definition and acquisition of its Integrated Support Facility computer resources (e.g., the software bench, called COMMANDS, and integrated test bed, called TATP). It was not clear to the supporting command personnel whether they would receive one, two, or even none of the current development test beds. Interoperability requirements do not seem to have been adequately addressed by the support activity. The procurement activity, contractor activity, and support activity were rated 3.0, 4.5, and 3.5 respectively, for an overall score of 3.67 for implementation methods.

g. The project procurement activity interfaces (external) have been plagued by politics, lack of interservice coordination, and system interoperability requirements. The procurement was planned as a joint service project, but there appears to be much independence on the part of the Air Force and Army participants. There have also been some problems with the Joint Program Office, the Mitre Corporation, and contractor interfaces. The procurement activity interface with the contractor has had some problems in establishing a good contractor QA program. The contractor has had some problems with the subcontractor. The support activity does not have a good interface definition with external project QA or configuration management elements. Support activity interfaces with the procurement activity have not been adequate to resolve the issue of acquisition of the test beds. The interfaces between the using and supporting command personnel appear to be improved through the evolution of the user/supporter baseline estimate. The procurement activity, contractor activity, and support activity were rated 2.0, 4.0, and 3.0 respectively, for an overall score of 3.0 for project (external) interfaces.

2.5.1.2 Software Configuration Management Evaluation Results.

a. The rationale behind the software configuration management evaluation results is discussed in this section.

b. The procurement activity configuration identification process has been very poor. In fact, the regulations (see AFR 65-3, MIL-STD-490, MIL-STD-483) requiring certain contractor code identification characteristics were not enforced (in fact were apparently waived). The reason for this waiver is that the Air Force Computer Program Identification Number (CPIN) assignment process is so complex, antiquated, and cumbersome that no one could complete the proper paperwork to get CPINs assigned. The effect is still a deficiency in the procurement activity. The contractor internal configuration identification method has been used. Generally, the procurement activity is responsible for assigning identifiers to software items at the CSCI level or above, and the contractor is responsible for assigning identifiers to software items below the CSCI level. Apparently, the contractor has done an excellent job of software item identification, baseline identification, and developmental/interim baseline identification (even though the format of the Version Description Document is poor). In addition, contractor identification of change requests, forms, etc. was very good. The support activity configuration identification is supposed to be overviewed in the CRISP and described in detail in the O/S CMP. The CRISP was inadequate and the O/S CMP was just being written. The procurement activity, contractor activity, and support activity were rated 2.0, 5.0, and 3.0 respectively, for an overall score of 3.33 for software configuration identification.

c. The procurement activity has generally relied on the contractor for configuration management, including control of the change process and the various baselines. The general requirement to control multiple variations of the (e.g. and Army) makes procurement control a more difficult task. The contractor has maintained good documentation and control board awareness of change activity and tracking even though the reporting aspect has not been visible enough to the procurement activity. The support activity has not yet adequately defined the support configuration control procedures which should be in the O/S CMP (being written at the time of this report). The procurement activity, contractor activity, and support activity were rated 3.0, 5.0, and 2.5 respectively, for an overall score of 3.5 for software configuration control.

d. The procurement activity, contractor activity, and support activity were all lacking capability in status accounting. Lack of automated tools, inadequate procurement requirements for contractor status accounting data, inadequate visibility of contractor status accounting data, support activity reliance on the upper level WR-ALC tracking forms (Form 75), and in general a lack of attention to the specification of status data all make this an area of concern for software supportability purposes. All activities were rated 3.0 for software configuration status accounting.

e. There was very little evidence of a configuration management audit capability. The formal procurement activity audits (FCA, PCA) are a form of configuration audits, but the important internal audit control function for each activity was limited to the usual review process. The support activity will have a requirement to audit the baselines, change process, status accounting procedures, and adherence of the configuration management process to established standards and conventions. The procurement activity, contractor activity, and support activity were rated 2.5, 4.0, and 2.0 respectively, for an overall score of 2.83 for configuration audit/review.

2.5.2 Software Product Maintainability Evaluation Results. The software documentation evaluation results for the average of the SICP and the NICP documentation were used as supplied by AFOTEC personnel. The source listing evaluation results for the average of five SICP modules and six NICP modules were used as supplied by AFOTEC personnel. The level 3 raw score values for modularity, descriptiveness, consistency, simplicity, expandability, and instrumentation were entered. The level 2 and level 1 results were averaged from level 3 and level 2 values, respectively.

2.5.3 Software Support Resources Evaluation Results. The software support resources evaluation results using all six evaluators and unweighted values were used as supplied by AFOTEC personnel. The level 3 values for personnel management, host systems, general facilities, etc were entered. The level 2 and level 1 results were averaged from level 3 and level 2 values, respectively.

2.5.4 General Software Supportability Evaluation Results.

a. Reference figure 2-8 for the following discussions. Three values in the figure refer to general software supportability evaluation results. The computed overall score of 3.80 is the average of the level 1 values: software life cycle process, software product maintainability, and software support resources.

b. The Software Supportability Confidence Assessment is a value between 0 (low) and 1 (high) which reflects the Software Test Manager/Deputy for Software Evaluation overall assessment of the evaluated software's supportability. In this pilot study, the BDM participant has completed this assessment and assigned a value of 0.70. This important value is only used during the evaluated risk regression equation update process (see the RAMSS User's Handbook, reference 1.4.6); it does not affect results of the current evaluation.

c. The evaluated software supportability risk value is computed from the five scores for software life cycle management, software product maintainability, personnel, support systems, and facilities, using a regression equation. The value of 0.55 represents the prediction that 55 percent of the block releases based on a workload estimate similar to the user/supporter baseline estimate cannot be completed without a workload or resource modification.

2.5.5. Software Supportability Risk Assessment.

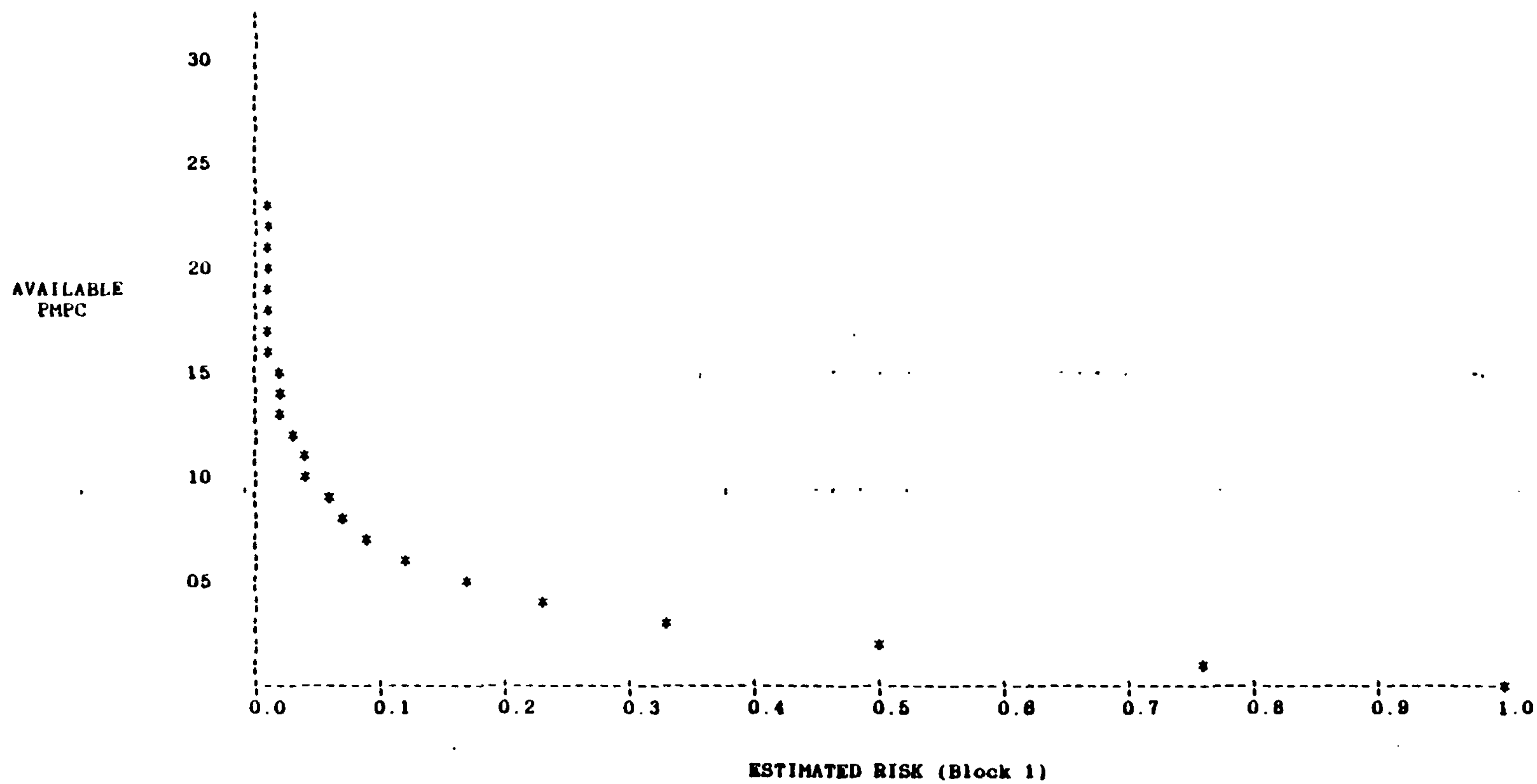
a. The software supportability evaluated risk is derived from the software supportability evaluation scores for software life cycle process, software product maintainability, support resource personnel, support resource systems and support resource facilities. A regression equation (see RAMSS User's Handbook, reference 1.4.6 and section 3.3) is used to evaluate the risk from these five evaluation scores.

b. The plots of the cumulative distribution function for each baseline estimate block release with various risk assessment values are shown in figures 2-9, 2-10, and 2-11. An overall summary of the software supportability risk assessment is shown in figure 2-12.

c. Based on a risk scale of 0.0 to 1.0 (with low risk values ≤ 0.20, medium risk values > 0.20 and ≤ 0.50, and high risk values > 0.50) the primary results of the risk assessment indicate that the overall evaluated software supportability risk is high, with the primary risk drivers being the software life cycle process and, at a lower level, the software configuration management process. Further analysis of these results is presented in section 2.6.

EVALUATION REPORT A5: PLOT OF PMPC VS. RISK

```
*****************************************************************
*                        SYSTEM PROFILE                         *
*                                                               *
* SWSYSID   :                         SWTYPE    : C-E           *
*                                                               *
* SYSTEM    :                         SUPPORTER: WR-ALC         *
*                                                               *
* SWSYSTEM :                          USER      : HQ-TAC        *
*****************************************************************
```



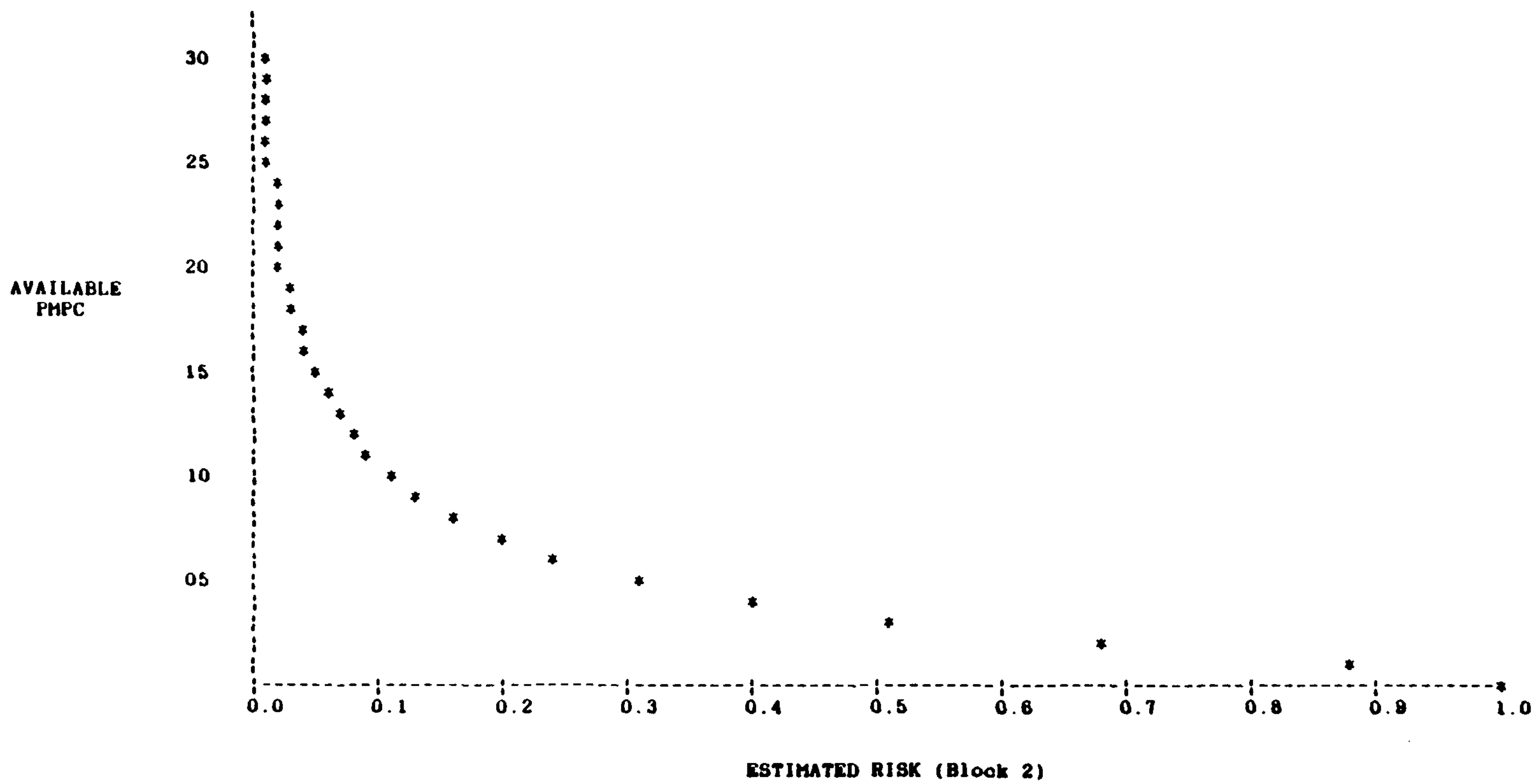
Evaluated Risk: 0.55

Available PMPC: 5.47
Estimated PMPC: 1.98
Estimated Risk: 0.14

Figure 2-9. Pilot Study Plot of Cumulative Distribution Risk Function (Block 1)

EVALUATION REPORT A5: PLOT OF PMPC VS. RISK

```
****************************************************************
*                        SYSTEM PROFILE                        *
*                                                              *
* SWSYSID   :                          SWTYPE    : C-E         *
*                                                              *
* SYSTEM    :                          SUPPORTER: WR-ALC       *
*                                                              *
* SWSYSTEM :                           USER      : HQ-TAC      *
****************************************************************
```



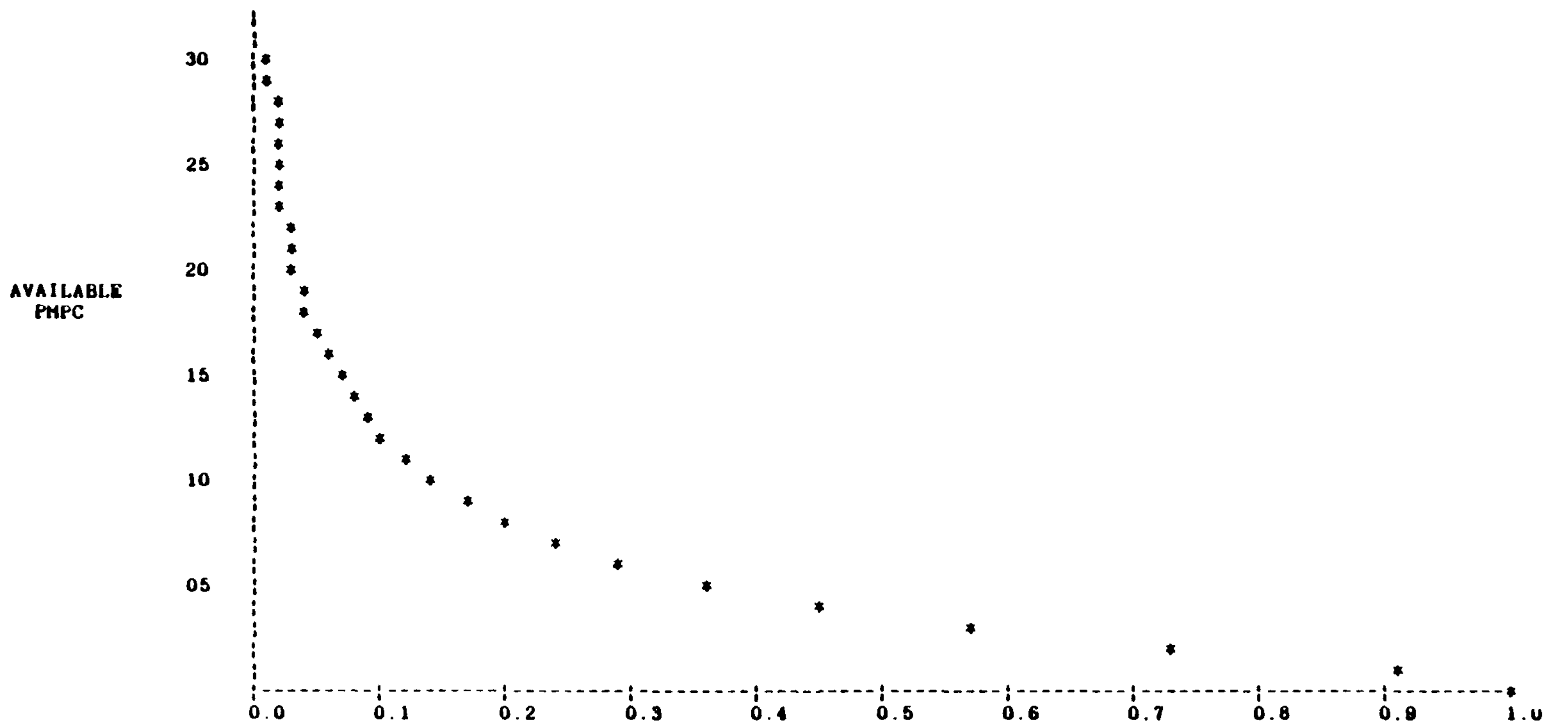
Evaluated Risk: 0.55

Available PMPC: 3.29
Estimated PMPC: 3.11
Estimated Risk: 0.48

Figure 2-10. Pilot Study Plot of Cumulative Distribution Risk Function (Block 2)

EVALUATION REPORT A5: PLOT OF PMPC VS. RISK

```
*************************************************************
*                       SYSTEM PROFILE                      *
*                                                           *
* SWSYSID   :                        SWTYPE    : C-E        *
*                                                           *
* SYSTEM    :                        SUPPORTER: WR-ALC      *
*                                                           *
* SWSYSTEM :                         USER      : HQ-TAC     *
*************************************************************
```



ESTIMATED RISK (Block 3)

Evaluated Risk: 0.55

Available PMPC: 3.29
Estimated PMPC: 3.57
Estimated Risk: 0.54

Figure 2-11. Pilot Study Plot of Cumulative Distribution Risk Function (Block 3)

```
EVALUATION REPORT A6: SUMMARY OF SOFTWARE SUPPORTABILITY RISK ASSESSMENT

              ****************************************************************
              *                          SYSTEM PROFILE                      *
              *                                                              *
              * SWSYSID   :                          SWTYPE    : C-E         *
              *                                                              *
              * SYSTEM    :                          SUPPORTER: WR-ALC       *
              *                                                              *
              * SWSYSTEM :                           USER      : HQ-TAC      *
              ****************************************************************

USER/SUPPORTER BASELINE ESTIMATE:

    SUPPORT STAFF:      15 Persons,  19% Dedicated, Avg.Skill level 3.00
                         9 Persons,  90% Dedicated, Avg.Skill level 3.00
    RELEASE SCHEDULE:  9.00 Month Block Release Cycle with  3.00 Month overlap
    BLOCK 1 CHANGE PROFILE: Total(  15); Type(  15,   0,   0); Complexity(   0,   0,  15); Priority(   0,   0,  15)
    BLOCK 2 CHANGE PROFILE: Total(  20); Type(  15,   5,   0); Complexity(   1,   4,  15); Priority(   1,   4,  15)
    BLOCK 3 CHANGE PROFILE: Total(  20); Type(  13,   7,   0); Complexity(   1,   6,  13); Priority(   1,   6,  13)

MAJOR FACTOR EVALUATION SCORES (THRESHOLD = 3.50, GOAL = 5.00):

    PROJECT MANAGEMENT:       3.47     LIFE CYCLE PROCESS: 3.32          SOFTWARE SUPPORTABILTY: 3.80
    CONFIGURATION MANAGEMENT 3.17

    DOCUMENTATION:            3.97     PRODUCT:            4.15
    SOURCE CODE:              4.32

    PERSONNEL:                3.53     SUPPORT RESOURCES   3.94
    SUPPORT SYSTEMS           3.72
    FACILITIES                4.58

MAJOR FACTOR RISK REDUCTION POTENTIAL:

    PROJECT MANAGEMENT:        0.21
    CONFIGURATION MANAGEMENT: 0.23

    DOCUMENTATION:             0.07
    SOURCE CODE:               0.06

    PERSONNEL:                 0.09
    SUPPORT SYSTEMS:           0.14
    FACILITIES:                0.00

OVERALL RISK ASSESSMENT  (THRESHOLD = 0.50, GOAL = 0.20):

    ESTIMATED (Block 1): 0.14     LOW
              (Block 2): 0.48     MEDIUM
              (Block 3): 0.54     HIGH

    EVALUATED: 0.55     HIGH
```

Figure 2-12. Pilot Study Summary RAMSS Results

2.6 ANALYSIS OF EVALUATION RESULTS.

This analysis will primarily be restricted to interpretation of the risk assessment reports generated by the RAMSS automated tool. These reports are as follows: A3: Major Factor Percentile Chart, A4: Major Factor Risk Reduction Chart, A5: Plot of Person Months per Change Versus Risk, and A6: Summary of RAMSS Results.

2.6.1 Report A3: Pilot Study Major Factor Percentile Chart.

a. The results of this report are shown in figures 2-13 (all systems) and 2-14 (C-E systems). Each of the major factors and level 1 criteria are compared with the distribution of all systems. The indicated percentages provide a relative understanding of how well the evaluated software compared with all other software systems. The higher the percentages, the better the evaluated software is relative to other software systems.

b. The percentile chart indicates the software life cycle process (the software configuration management factor), was relatively the worst. The software product (the source listings factor) was the best. As a guideline, if the score is $\geq$ 0.75, it is high; if it is below 0.25, it is low. Deficiencies might be noted for scores below 0.25.

2.6.2 Report A4: Pilot Study Major Factor Risk Reduction Chart.

a. The results of this report are shown in figure 2-15. The major factor and criteria risk impact is determined by computing the difference in the evaluated risk for the actual versus a perfect score. This difference is then plotted as in figure 2-15. The interpretation of these differences is that they represent the amount the evaluated risk could be reduced if the given factor or criteria were to be improved as much as possible (in other words, given a

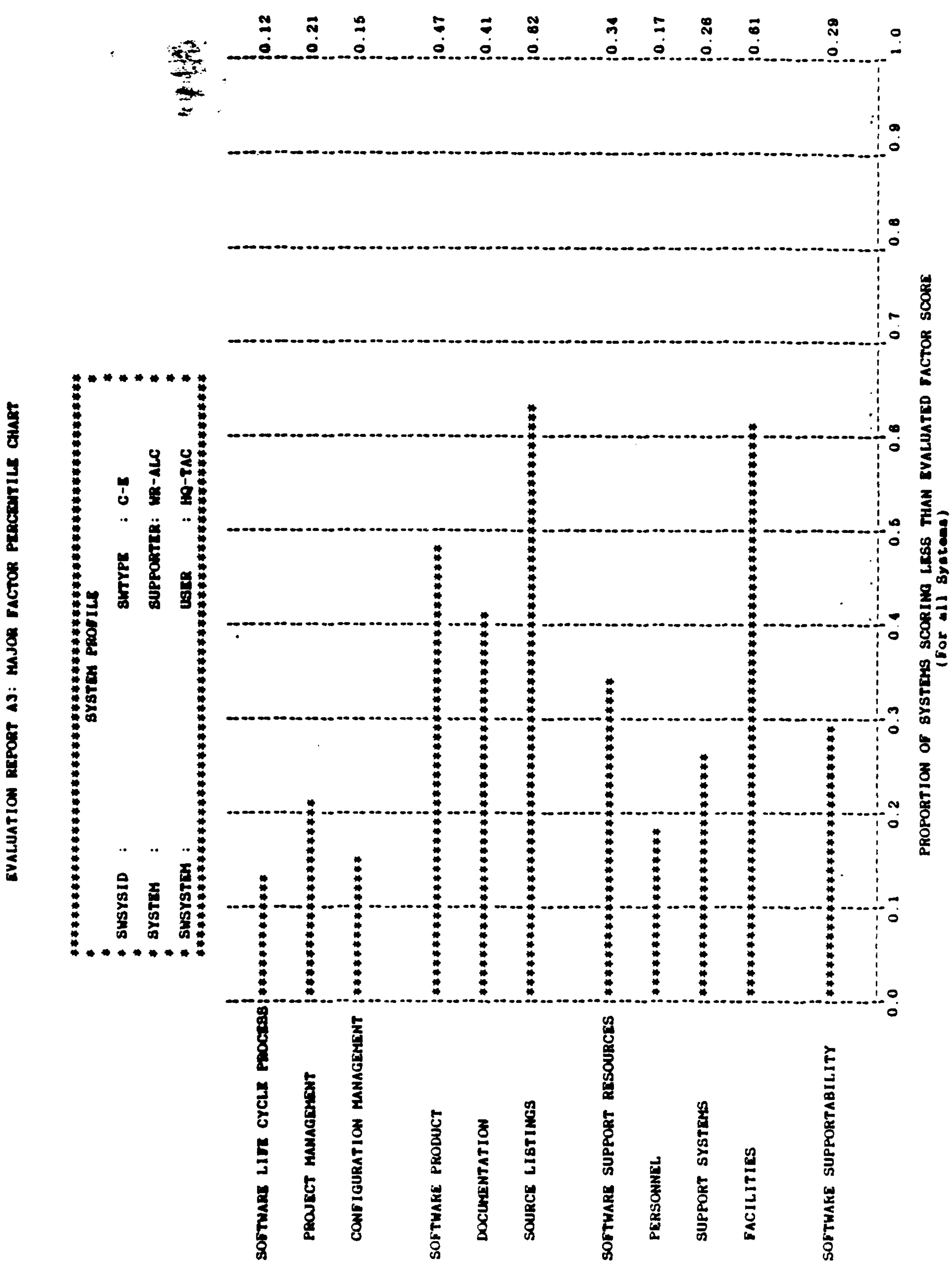


Figure 2-13. Major Factor Percentile Chart (All Systems)

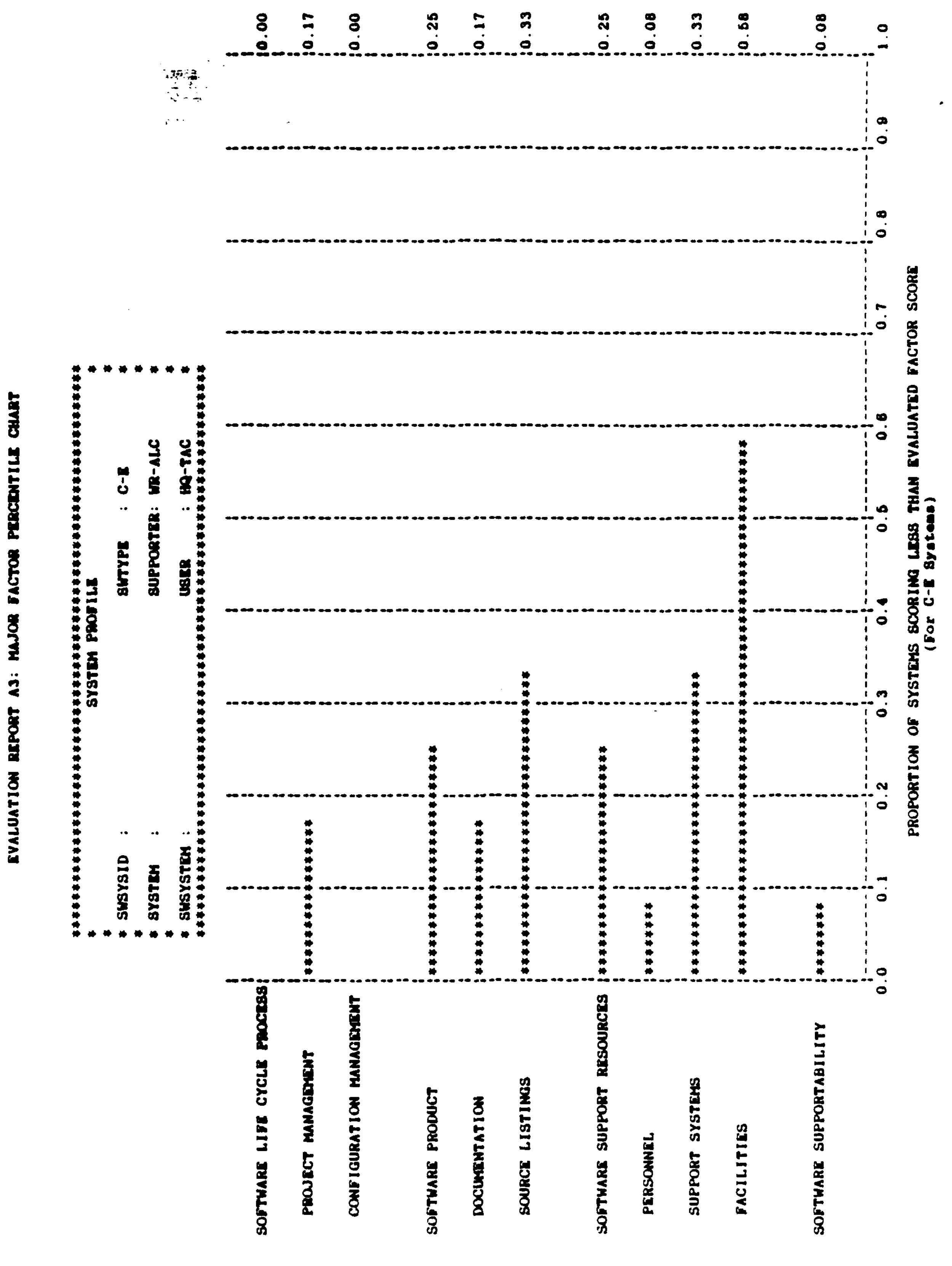


Figure 2-14. Major Factor Percentile Chart (C-E Systems)

EVALUATION REPORT A4: MAJOR FACTOR RISK IMPACT CHART

SYSTEM PROFILE	
SWSYSID :	SWTYPE : C-E
SYSTEM :	SUPPORTER: WR-ALC
SWSYSTEM :	USER : HQ-TAC

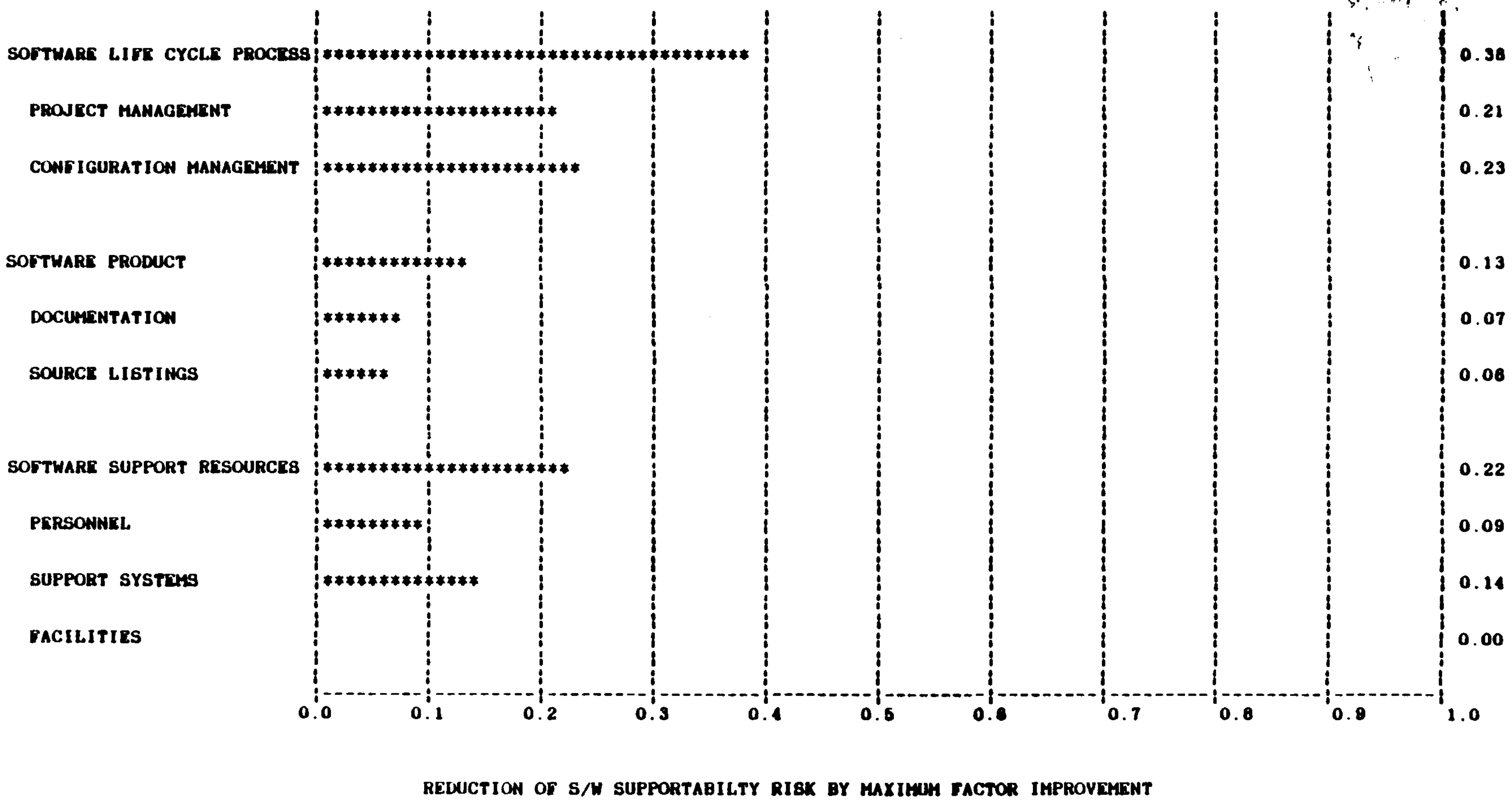


Figure 2-15. Major Factor Risk Reduction Chart

perfect score of 6.0). Thus, the factor or criteria which could have the most impact on reduction of risk will have the longest plot of asterisks. The plot of figure 2-15 thus gives a ranking of all the major factors and criteria according to impact on the risk reduction of software supportability for the Software (actually, only the evaluated NICP and SICP software). The risk values are computed from the evaluated risk regression equation.

b. Of course, there will be tradeoffs as to which factors and criteria with the most risk impact are viable candidates for improvement, and which lower-level characteristics of those viable candidates can be improved. The course of action would be to look at the low scoring characteristics for each viable candidate factor/ criteria, estimate a reasonable improvement in the score, recompute the factor/criteria score, and enter into the risk regression equation to determine the amount of reduction in the risk. This process may have to be reaccomplished several times to determine the optimum tradeoff benefits.

c. From figure 2-15, the candidates for possible risk reduction are the software configuration management factor and the software project management factor. Focusing on the software configuration management major factor, the question is whether there are lower-level characteristics which can be improved, and if so, by how much. Noting that the is in the fifth year of what appears to be a 10-year full scale development effort (with PMRT in 1990), it appears there is still plenty of time to effect significant improvement in all three activities: procurement, contractor, and support.

d. Major improvements in the procurement activity would occur if: current procedures would be documented; the baseline identification function were to be properly managed; automated tool support were implemented which could be transitioned to the support activity; and significant planning and preparation for product acceptance through the FCA and PCA for the product baseline were accomplished. Improvement in the procurement activity from 2.63 to 4.25 is possible. The contractor could improve some by automating status accounting functions and improving the audit/review process. Not much improvement is anticipated. The contractor activity might improve from 4.25 to 4.5. The support activity has the best chance of improvement by: developing an agreed on and high quality O/S CMP; updating the CRISP to include proper high level detail consistent with the O/S CMP; adopting rigorous internal procedures and standards; writing and adhering to an internal configuration management plan; and working with the procurement activity to effect an automated configuration management system for baseline software documentation and code control, and change status accounting and reporting. Improvement in the support activity from 2.63 to 5.0 is possible. Overall this represents an improvement in the configuration management major factor from 3.17 to 4.58. This would represent a reduction in overall evaluated software supportability risk from 0.55 to 0.43.

2.6.3 Report A5: Pilot Study Plot of Cumulative Distribution Risk Function.

a. The results of this report are shown in figures 2-9, 2-10, and 2-11. The primary use of these plots is to visually display the software supportability risk versus the workload of available person months per change (PMPC). The two figures are plots for each baseline estimate block release for the evaluated system. From these plots the analyst can get a feel for the variances between the risk (estimated and evaluated), and among the person months per change workloads (available, estimated, evaluated). See the Glossary for definitions of key terms used above.

b. If the evaluated risk is lower than the estimated risk, that says the quality of the life cycle process, software products, and support resources may overcome some of the estimated risk based only on workload and expected change profile. If both evaluated and estimated risk are low (e.g., below 0.2), then the software supportability risk for the system can be considered low. If both evaluated and estimated risk are high (e.g., above 0.5), then the software supportability risk for the system can be considered to be high.

c. The analyst can perform "what-if" sensitivity analysis with the cumulative distribution plots and the risk values. For example, the evaluated risk is derived from the software supportability evaluation scores. Those scores were based on the baseline estimate. If the baseline estimate were to be changed (i.e., increase personnel, decrease change profile workload), then theoretically the software supportability evaluation would have to be conducted again against the new baseline. Since this is not practical, an alternate approximation is suggested. Using the cumulative distribution plot (e.g., figure 2-10 for block release 2), the analyst should plot the evaluated risk value (0.55) and determine the evaluated person months per change (approximately 2.76). Then, use the RAMSS tool to construct a new cumulative distribution function based upon the new block release 2 baseline. Using the old 2.76 evaluated person months per change plotted against the new cumulative distribution function would give the approximate evaluated risk against the new baseline estimate. The change in the evaluated risk (reduction in this case) could then be noted for sensitivity purposes and risk reduction analysis.

2.6.4 Report A6: Pilot Study Summary of RAMSS Results.

a. The results of this report are shown in figure 2-11. This report presents a concise summary of the evaluation scores, major factor risk impact, estimated and evaluated risk, and ratings of the software supportability risk values as HIGH (above threshold), MEDIUM (threshold to goal), or LOW (below goal). The threshold and goal values are subjective at this time, but represent:

(1) Threshold = 0.5 means 50 percent of the time the user/supporter baseline workload estimate will not be met.

(2) Goal = 0.2 means 20 percent of the time the user/supporter baseline workload estimate will not be met.

b. Even with these values, it does not necessarily mean that missing the estimate will have much impact. If a scheduled release misses by a few days, it may or may not be a large impact. No risk impact functions (more correctly called utility functions) have been derived as part of this effort. However, if one were to consider negative events to be normally distributed with catastrophic and very minor impact as the respective boundary conditions, then some estimate of mean and standard deviation could be projected for each specific risk agent and a utility function generated. For example, this might mean that given a risk of 0.4, only 1 percent of the negative events are catastrophic or with a risk of 0.5 as many as 25 percent of the negative events are catastrophic. The investigation of utility functions is beyond the scope of the current analysis effort.

2.7 LESSONS LEARNED.

a. The following list summarizes the lessons learned during the pilot study application of RAMSS to the software.

(1) The user/supporter baseline estimate (USBE) was able to be derived, but required some reasonable "guesses" based

upon maintenance release data. The RAMSS support tool will help this process, but some continual training of AFOTEC personnel will be required.

(2) The main benefit of the USBE was in stimulating the interaction and discussion between using and supporting commands, and within the using command. It is recommended that the USBE, in agreement form, be present in a software support document (TPO, CRISP, or CRLCMP).

(3) The USBE was not a major factor in answering the individual evaluation questions.

(4) The SLCP evaluation cannot be done in the same manner as the other evaluations. For credibility, it is essential to capture the life cycle process characteristics over time and create an historical base upon which the SLCP questions can be answered. However, it should be possible to interact with system experts during the life cycle to capture this history using the SLCP questions (see reference 1.4.7) as a checklist.

(5) The use of the RAMSS tool will aid the interpretation of the risk assessment results. There are several "what if" functions that can be done. For example, in trying to pinpoint specific characteristic risk reduction, it is possible to determine the "new" risk given several "temporary" changes to characteristics scores (see report A2 discussion in appendix B).

(6) One important side effect of the USBE evolution process is the data gathered from both using and supporting command personnel concerning areas of risk. These areas of risk can be investigated by AFOTEC personnel for

potential impact upon the system OT&E as well as the software portion of the OT&E.

(7) The using and supporting command personnel were very cooperative during the pilot study. They seemed to appreciate the opportunity to participate in the specification of the USBE. The using command had some reservations initially about AFOTEC's role in this area. Reluctance to "sign-up" to agreed on USBE values may occur for some systems.

(8) The development effort has several "generic" life cycle process flaws which have been observed across many systems:

a) The full scale development schedule of 27 months defined in 1980 was much too ambitious. Current projections are for PMRT in 1990.

b) Functional expectations changed from the prototype demonstration.

c) Interoperability requirements with other services were (are) a source of problems.

d) Planning for computer support resources during the post-deployment phase has been very poor. Generally, very little priority is given to this function.

e) Organizational centralization of responsibility and consistency of personnel over the project life cycle has been poorly managed.

f) Configuration management plans, procedures, and automated tool support are inadequate.

g) Security concerns (e.g., classified software documentation and source listings) have not been properly addressed. Solution is frequently to declassify information (perhaps arbitrarily).

h) Procurement activity understanding of the deliverable requirements, as reflected in the RFP/CDRL/etc., has been inadequate in the area of computer resources, test support, and quality assurance.

b. There is a significant amount of project management and configuration management which is being done, but not being properly incorporated into the proper planning, specifications, and other documents. For example, the support activity personnel knew much more information concerning the plans, organization structure, test strategies, configuration control, personnel allocations, and facility layout than was contained in the TEMP, CRISP, or O/S CMP. Apparently, these documents are not having as much effective use as is possible.

III. Refinements to RAMSS

SECTION III

REFINEMENTS TO RAMSS

3.1 INTRODUCTION.

The baseline for the RAMSS is contained in volumes 1 and 2 of the report "Software Supportability Risk Assessment in OT&E: Historical Baselines for Risk Profiles," (see reference 1.4.5). There have been several refinements to the methodology, procedures, and statistical analysis results represented in that baseline report. This section discusses the major refinements in some detail. The complete RAMSS Evaluator's Guide, which integrates these refinements, is presented in appendix B of this document.

3.2 TRANSFORMATION OF HISTORICAL EVALUATION DATA.

a. The RAMSS baseline report (reference 1.4.5) contained a data base of evaluation on 81 software systems using the new software supportability hierarchy across the three criteria: software life cycle process, software product maintainability, and software support resources. The hierarchy evaluation scores were obtained at each of the levels (3, 2, 1, 0) of the hierarchy (reference figure 2-7).

b. For purposes of continued application of this historical data for future AFOTEC evaluations, a new evaluation data base has been generated using the level 3 evaluation values. The values were converted from a -50 to +50 scale, to the AFOTEC 1-6 scale. The level 3 values were averaged to obtain level 2 values and so forth until evaluation scores at all levels were computed. The old evaluation data base level 0 score (ASUPPORT for overall software supportability) has been transformed to a new variable ACONFID which has a value between 0 (low) and 1 (high) representing the overall software supportability confidence rating. The transformation equation is:

$$\text{ACONFID} = \frac{50 + \text{ASUPPORT}}{100}$$

The old data base risk variable, ARISK, was determined to be too erratic for further use. The concept of "confidence" seemed to be easier for an evaluator to assess consistently than that of "risk", even though for our purposes the general software supportability risk would be simply 1-ACONFID. It is this risk value against which the software supportability evaluated risk regression equation has been derived (see section 3.4) using the evaluation factors described in section 3.3. This use of ACONFID in the derived risk regression equation is the only use of this variable, but it enables AFOTEC to maintain a reasonably current equation for computing the evaluated software supportability risk for an evaluated software system using an extensive historical data base. As the data base evolves and becomes more accurate and larger, the regression equation should be more accurate as well.

c. Further specification of the historical evaluation data as it has been transformed is described in the RAMSS User's Handbook (reference 1.4.6) along with all the data base information which is a part of this risk assessment methodology.

3.3 NEW SUPPORTABILITY FACTORS.

a. The RAMSS baseline report (reference 1.4.5) presented an analysis approach (in section 4.4.2.1) to determine the grouping relationship of the 44 supportability rating variables used on the data collection survey form. This analysis approach is called factor analysis. After studying the data, 7 variables were eliminated and the remaining 37 were used. The factor loadings resulting from that analysis are shown in figure 3-1. The interpretation of those results is shown in table 3-1.

ROTATED FACTOR LOADINGS

		FACTOR 1	FACTOR -2	FACTOR 3	FACTOR 4	FACTOR 5	FACTOR 6
APOOC	3	0.138	0.743*	0.091	0.369	0.207	0.202
APOOCMGD	4	0.189	0.541*	0.239	0.654*	0.019	0.013
APOOCDES	5	0.289	0.729*	-0.003	0.024	0.297	0.192
APOOCCON	6	0.308	0.599*	0.264	0.422	0.206	-0.161
APOOCSIM	7	0.153	0.689*	0.002	0.057	0.055	0.285
APOOCEXP	8	-0.111	0.476	0.341	0.459	-0.083	0.084
APSRC	10	0.358	0.726*	0.259	0.172	0.176	0.026
APSRCMOD	11	0.236	0.566*	0.222	0.608*	-0.054	-0.141
APSRCDES	12	0.364	0.767*	-0.010	0.021	0.215	-0.057
APSRCCON	13	0.225	0.717*	0.152	0.242	0.223	-0.177
APSRCSIM	14	0.018	0.810*	0.014	-0.017	0.014	0.170
APSRCEXP	15	-0.148	0.670*	0.522*	0.027	-0.021	0.085
APROCUCT	17	0.204	0.820*	0.246	0.159	0.089	0.213
AEPER	18	0.370	0.181	0.815*	0.207	0.130	0.085
AEPERMAN	19	0.510*	-0.032	0.542*	0.341	0.238	0.011
AEPERTEC	20	0.224	0.128	0.855*	0.117	0.092	0.036
AEPERSUP	21	0.286	0.169	0.802*	0.296	0.074	0.049
AESYS	23	0.142	0.352	0.408	0.087	0.282	0.588*
AESYSHOS	24	0.011	0.284	0.036	0.075	0.218	0.844*
AEFAC	29	0.168	0.134	0.057	-0.039	0.937*	0.117
AEFACCPP	30	-0.057	0.123	0.061	0.300	0.890*	-0.047
AEFACENV	31	0.272	0.219	0.103	-0.054	0.771*	0.329
AENVIRON	32	0.337	0.240	0.295	0.058	0.768*	0.143
AMCON	33	0.621*	0.215	0.193	0.623*	0.023	-0.127
AMCONIDE	34	0.400	0.153	0.083	0.733*	0.158	0.000
AMCONSTA	35	0.864*	0.103	0.025	0.134	-0.036	-0.144
AMCONCON	36	0.804*	0.231	0.127	0.334	0.119	-0.100
AMCONAUD	37	0.876*	0.101	-0.013	0.063	0.054	-0.122
AMMAI	38	0.708*	0.270	0.351	0.301	0.232	0.223
AMMAIPLA	39	0.693*	0.137	0.357	0.271	0.254	0.179
AMMAICRG	40	0.212	-0.026	0.219	0.736*	0.103	0.300
AMMAICES	41	0.663*	0.360	0.389	0.151	0.133	0.178
AMMAICOD	42	0.530*	0.466	0.423	0.013	0.185	0.160
AMMAITES	43	0.754*	0.243	0.189	-0.079	0.101	0.273
AMMAIINT	44	0.710*	0.147	0.307	0.148	0.120	0.101
AMANAGE	45	0.725*	0.144	0.189	0.390	0.278	0.257
ASUPPCRT	46	0.515*	0.450	0.418	0.130	0.076	0.251

Figure 3-1. Baseline Report Factor Analysis Results

b. The factor analysis results were very encouraging since all the factors except "organization" were elements in AFOTEC's software supportability evaluation hierarchy. The "organization" factor seems to cross several of the hierarchy elements. These factor results were satisfactory enough to consider using a five-factor model (all but the "organization" factor) for regression analysis. Computational complexity of the factor analysis model justified a simplification of the five-factor model so that the factor values are simply the average of the factor's lower level characteristic values. Thus, the new software supportability factors for use in regression analysis are shown in table 3-2, and the factor values are obtained from a simple cumulative average rather than by the complicated factor analysis computation. The significant factor analysis values (indicated by an asterisk in figure 3-1) justify the use of the evaluation hierarchy characteristic values.

Table 3-1.
Baseline Report Supportability Factors

INTERPRETATIONS OF SUPPORTABILITY FACTORS	
FACTOR NUMBER	INTERPRETATION
1	SUPPORT MANAGEMENT
2	PRODUCT
3	PERSONNEL
4	ORGANIZATION
5	FACILITIES
6	SUPPORT SYSTEMS

86-0360-TR-W-04

Table 3-2.
New Software Supportability Factors

INTERPRETATIONS OF SUPPORTABILITY FACTORS	
FACTOR NUMBER	INTERPRETATION
1	SOFTWARE SUPPORT LIFE CYCLE PROCESS
2	SOFTWARE PRODUCT
3	SUPPORT PERSONNEL
4	SUPPORT SYSTEMS
5	SUPPORT FACILITIES

86-0360-TR-W-05

3.4 REGRESSION EQUATION FOR EVALUATED SUPPORTABILITY RISK.

a. The mathematical model for regression analysis is

$$Y = b_0 + b_1X_1 + \ldots + b_5X_5 + e$$

where

Y = the transformed general software supportability risk rating

X_i = the score for the ith factor

e = a random component

and the regression coefficients b_0, b_1, ..., b_5 are parameters to be estimated. The X_i values are computed for the five factors described in section 3.3. The general software supportability risk rating is derived from the historical evaluation data base variable ACONFID. The ACONFID value is the evaluator's overall confidence in the supportability of the software based upon all possible software supportability factors and a baseline estimate of the software change profile. The value is obtained from an evaluator (probably the Software Test Manager) independently of the other software supportability evaluations.

b. The computation of the evaluated software supportability risk thus follows from the mathematical model:

R = risk = 1-ACONFID $0 \le R \le 1$

$\hat{R}$ = predicted risk (evaluated software supportability risk)

$$\hat{R} = \left[\frac{1}{1 + e^{-\hat{L}}} - \frac{a}{2}\right](1 - a)^{-1} \qquad a = 0.02$$

$$R' = R(1 - a) + \frac{a}{2}$$

where

$$\hat{L} = \ln\left(\frac{R'}{1 - R'}\right)$$

$$= b_0 + b_1(\text{APRODUCT}) + b_2(\text{AEPER}) + b_3(\text{AESYS})$$

$$+ b_4(\text{AEFAC}) + b_5(\text{AMANAGE})$$

The constants b_i are the regression equation coefficients.

c. The BMDP (reference 1.4.10) statistical regression package results for this model are shown in figure 3-2. The equation for $\hat{L}$ thus becomes:

$$\hat{L} = 4.90401 - 0.29131\ (\text{APRODUCT}) - 0.15600\ (\text{AEPER})$$
$$- 0.25120\ (\text{AESYS}) + 0.04294\ (\text{AEFAC})$$
$$- 0.66174\ (\text{AMANAGE})$$

PAGE 3 BMDP1R L(RISK) VS. SUPPORTABILITY FACTORS
REGRESSION TITLE IS
L(RISK) VS. SUPPORTABILITY FACTORS

0 DEPENDENT VARIABLE. . . . . 15 LRISK
TOLERANCE 0.0100
ALL DATA CONSIDERED AS A SINGLE GROUP
0MULTIPLE R 0.8086 STD. ERROR OF EST. 0.6239
MULTIPLE R-SQUARE 0.6539

ANALYSIS OF VARIANCE

	SUM OF SQUARES	DF	MEAN SQUARE	F RATIO	P(TAIL)
REGRESSION	38.2376	5	7.6475	19.650	0.0000
RESIDUAL	20.2382	52	0.3892		

VARIABLE		COEFFICIENT	STD. ERROR	STD. REG COEFF	T	P(2 TAIL)	TOLERANCE
INTERCEPT		4.90401					
APRODUCT	4	-0.29131	0.13090	-0.226	-2.225	0.0304	0.64641
AEPER	5	-0.15600	0.12403	-0.136	-1.258	0.2141	0.56613
AESYS	6	-0.25120	0.12905	-0.181	-1.946	0.0570	0.77162
AEFAC	7	0.04294	0.10015	0.043	0.429	0.6699	0.66939
AMANAGE	11	-0.66174	0.14256	-0.509	-4.642	0.0000	0.55331

Figure 3-2. Results for Regression Analysis (Transformed) Risk Versus Supportability Factors

d. The sequence of computations to determine the evaluated software supportability risk given the evaluation values for APRODUCT, AEPER, AESYS, AEFAC, and AMANAGE is illustrated by the following examples:

Suppose:

APRODUCT (Product Maintainability) = 4.15

AEPER (Personnel Resources) = 3.53

AESYS (System Resources) = 3.72

AEFAC (Facility Resources) = 4.58

AMANAGE (Life Cycle Process) = 3.32

then:

$$\hat{L} = 4.90401 - 0.29131\ (4.15) - 0.15600\ (3.53) - 0.25120\ (3.72) + 0.04294\ (4.58) - 0.66174\ (3.32)$$

$$= 0.20962$$

$$\hat{R} = \left[\frac{1}{1 + e^{-0.20962}} - \frac{.02}{2}\right] (1 - .02)^{-1}$$

$$= 0.55$$

e. The only anomalous aspect of this regression equation is the plus sign of the AEFAC coefficient. This would seem to imply that the better the AEFAC the higher the risk. Actually, the AEFAC factor is not significant, as can be determined from the P(2 TAIL) column in figure 3-2. The AEFAC coefficient, even though it is positive, is very small. The AEFAC factor is retained to maintain parallelism with the AFOTEC software supportability evaluation hierarchy. The regression equation will evolve over time as more evaluations are performed by AFOTEC, and data is added to the historical evaluation data base.

3.5 COMPUTATION OF SUPPORTABILITY FACTOR RISK REDUCTION.

a. An important aspect of risk analysis is to determine which software supportability criteria/major factors have the most "impact" upon the risk. A simple method was derived to obtain some perspective of the maximum possible reduction in evaluated software supportability risk due to each of the software supportability criteria and major factors. Those criteria/major factors which could potentially effect the most reduction in risk could be considered to be the risk "drivers."

b. The method of computing the risk reduction is as follows:

(1) Compute the evaluated software supportability risk R_E from the regression equation using the five factor evaluation scores.

(2) For each software supportability criterion and major factor, determine the risk under the assumption of maximum criterion/major factor improvement (i.e., an evaluation score of 6.0).

(3) Compute the difference in risk $R_E - R_i$ for each criteria/factor i.

(4) The criteria/factors with the largest computed difference, i.e., largest potential reduction in risk, are considered to be the risk drivers.

c. As an example, the data from the example in section 3.4 resulted in an evaluated software supportability risk of 0.55. The following potential risk reductions are computed:

APRODUCT (4.15 - 6.00)	Risk Reduction = 0.55 - 0.42 = 0.13
AEPER (3.53 - 6.00)	Risk Reduction = 0.55 - 0.46 = 0.09
AESYS (3.72 - 6.00)	Risk Reduction = 0.55 - 0.41 = 0.14
AEFAC (4.58 - 6.00)	Risk Reduction = 0.55 - 0.57 = -0.02
AMANAGE (3.32 - 6.00)	Risk Reduction = 0.55 - 0.17 = 0.38
SUPPORT RESOURCES (3.94 - 6.00)	Risk Reduction = 0.55 - 0.33 = 0.22

These [illegible]lts clearly show that the AMANAGE (Life Cycle Process) criteri[illegible] the most impact, followed by SUPPORT RESOURCES, and then APRODUC[illegible] In a similar manner, the major factor components of AMANAGE and APRODUCT can be included to determine the potential reduction in risk due to an improvement in those major factors. The results can be compared against the AEPER, AESYS, and AEFAC risk reduction results to determine major factor risk drivers.

d. The RAMSS analysis report "A4: MAJOR FACTOR RISK REDUCTION CHART" gives a pictorial representation of the potential risk reduction results. Examples of this report are in section 2, appendix B, and in the RAMSS User's Handbook (reference 1.4.6). Further analysis can be easily conducted to determine which criteria/major factors/characteristics can feasibly be improved across the evaluation scores. The potential reduction of the evaluated supportability risk can then be computed. For the example above, suppose such an analysis were to focus upon the software life cycle process (AMANAGE), since this factor appears to have the most impact, and it were determined that a realistic improvement in the characteristics would raise the evaluation score from 3.32 to 4.58. Then the evaluated software supportability risk would drop from 0.55 to 0.35, a substantial reduction. Further analysis may be performed by the STM/OSE to identify other areas where improvement would significantly raise the overall evaluation score.

3.6 REGRESSION EQUATION FOR BASELINE ESTIMATED WORKLOAD.

a. During the analysis effort for the RAMSS baseline report (reference 1.4.5) there was not sufficient time to determine whether important relationships existed among the data collected for the maintenance releases. In particular, it was anticipated that the resources expended in person months for a given release might be dependent upon: the skill level of the personnel; the distribution of changes across type, complexity, and priority; and the functional nature of the software. Other parameters such as number of source lines (total) and percentage of high level language source lines (total) might also affect the resources expended. Parameters such as the percentage of source lines changed and percentage of changed source lines which are high level language could not be collected as part of the data collection effort.

b. A regression analysis using a model similar to the model described in section 3.4 has been done, and the results are significant enough to incorporate into the RAMSS. The following regression equations have been derived to predict the person months per change workload required for a given profile of change requests on a system whose software is of type OFP, CE, EW, ATD, ATE, SIM, or SUP, and whose support personnel have a certain average skill level. The historical data is reasonably accurate, but there is hope that improved maintenance release data in the future might improve our understanding of these macro relationships.

c. The regression model equations are:

PMPC = Person months per change request workload

$\hat{PMPC}$ = predicted workload

$$\hat{PMPC} = e^{\hat{L}}$$

where

$$\hat{L} = b_0 + b_1(AVGSKILL) + b_2(PTCORR) + b_3(PCLOW) + b_4(PCHIGH) + b_5(PPNORM) + \sum_{i=6}^{10} b_i(TYPE_i)$$

AVGSKILL = average skill (1 -low to 5-high) of support personnel

PTCORR = percentage of change requests which are corrections

PCLOW = percentage of change requests which are low complexity

PCHIGH = percentage of change requests which are high complexity

PPNORM = percentage of change requests which are normal priority

$TYPE_i$ = 1 if system type is same as $TYPE_i$
0 otherwise

Five types have explicit coefficients. The INTERCEPT constant is the implicit coefficient for both SUP and SIM. The results of the regression analysis including the coefficients b_i are shown in figure 3-3. The significant variables are PCHIGH, PPNORM, AVGSKILL, and in some respect, the system functional type (ATD, ATE, ...). Because the addition of more maintenance release data to the historical maintenance release data base could provide new insights into the parameters, the complete set of parameters shown in figure 3-3 will be retained in this initial regression model.

PAGE 3 BMDP1R SW TYPE DUMMY VARIABLES PLUS COVARIATES
REGRESSION TITLE IS
SW TYPE DUMMY VARIABLES PLUS COVARIATES

0 DEPENDENT VARIABLE. . . . . 38 LN(PMPC)
TOLERANCE 0.0100
ALL DATA CONSIDERED AS A SINGLE GROUP
0MULTIPLE R 0.5774 STD. ERROR OF EST. 0.9509
MULTIPLE R-SQUARE 0.3334

ANALYSIS OF VARIANCE

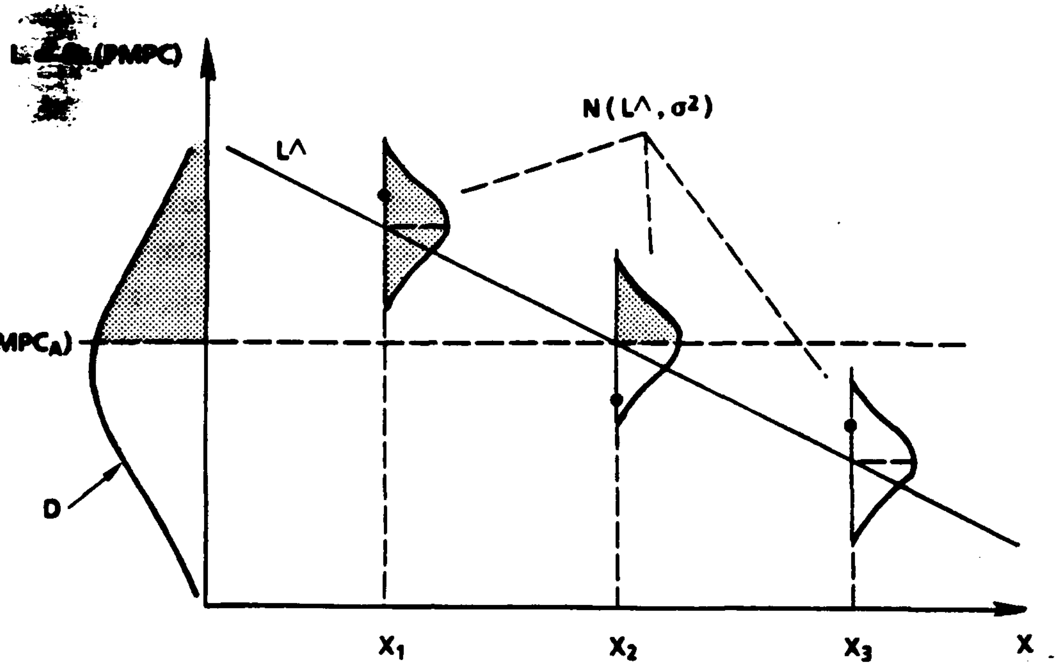
	SUM OF SQUARES	DF	MEAN SQUARE	F RATIO	P(TAIL)
REGRESSION	63.3254	10	6.3325	7.003	0.0000
RESIDUAL	126.5911	140	0.9042		

VARIABLE		COEFFICIENT	STD. ERROR	STD. REG COEFF	T	P(2 TAIL)	TOLERANCE
INTERCEPT		0.07083					
AVGSKILL	7	0.54837	0.13224	0.323	4.147	0.0001	0.78516
PTCORR	29	-0.02948	0.39988	-0.007	-0.074	0.9413	0.52019
PCLOW	32	0.21815	0.40660	0.052	0.537	0.5924	0.51666
PCHIGH	34	2.09563	0.78857	0.255	2.658	0.0088	0.51551
PPNORM	35	-1.57228	0.61743	-0.208	-2.546	0.0120	0.71313
ATD	39	-0.50845	0.49983	-0.081	-1.017	0.3108	0.74867
ATE	40	0.14413	0.43891	0.033	0.328	0.7431	0.46024
C-E	41	0.35255	0.29391	0.155	1.200	0.2324	0.28386
EW	42	0.85802	0.46308	0.171	1.853	0.0660	0.55656
OFP	43	0.04114	0.29275	0.017	0.141	0.8884	0.33399

Figure 3-3. Results for Regression Analysis (Transformed) Person Months Per Change Versus Maintenance Release Profile Data

d. In order to obtain the estimated risk, a refinement of the technique used in the RAMSS baseline report (reference 1.4.5) has been derived. In the baseline report a simple counting technique was employed to determine the estimated risk. The historical sample of number of systems releases within a certain range of person months per change was viewed as a probability density function. The estimated risk for the system being evaluated was then computed as the area under the curve greater than or equal to the available person months per change as computed from the user/supporter baseline estimate. Two distributions were of interest: all system release data points, and all system release data points for systems of the same type as the system being evaluated. Thus, there would be two estimated risks which could be computed. The two distributions approximated a normal distribution with some universal mean and standard deviation.

e. From the regression equation for the estimated person months per change, a more accurate refinement of the computation for estimated risk can be derived in which the covariates of the regression equation are significant. It is assumed that the historical values for available person months per change are normally distributed with mean (the estimated person months per change) and variance (the square of the standard error of estimation from the regression analysis is a best estimate). This distribution of the available person months per change (L_1) about the regression line ($\hat{L}$) for one of the covariates (X) is illustrated in figure 3-4. If all such regression lines for all covariates were flat (zero slope), then the resulting one distribution (shown on the left of figure 3-4) would correspond to the baseline report historical sample distribution for all systems. Thus, this refinement results in a family of distribution functions oriented about the regression equation.



SHADED AREAS REPRESENT RISK

Figure 3-4. Illustration of Estimated Risk Distribution Functions

f. Given the situation shown in figure 3-4, it becomes possible to refine the current risk estimation procedure by taking advantage of the regression model. Recall, in the old procedure, an available PMPC value, $PMPC_A$, is compared to the distribution of sample PMPC values across the releases of all systems or systems of one type, and risk is estimated as the proportion of sample releases having PMPC values greater than or equal to $PMPC_A$. This risk estimate is illustrated as the shaded area under the distribution curve labelled D that appears against the ℓn(PMPC) axis of figure 3-4.

g. Use of a sample distribution like D to estimate risk is fine in the absence of a more detailed model to represent PMPC. Once a more sophisticated model can be built, though, that model should be exploited to estimate risk. In the regression model used here, PMPC is related to such variables as proportion of high complexity changes; therefore, a risk estimate should also be related to those variables. Figure 3-4 shows that the regression approach may yield very different risk estimates than the sample distribution approach. Note that estimated risk (the shaded area under the curves labelled $N(\hat{L},\sigma^2)$ depends on the value of the generic covariate X. If $X = X_1$, estimated risk is nearly 1; if $X = X_3$, estimated risk is nearly 0; if $X = X_2$, risk is 1/2, a value close to that estimated via the sample distribution. The sample distribution yields one risk estimate that is not influenced by X values and is therefore misleading in light of the regression model.

h. Tables 3-3 and 3-4 compare risk estimates based on the two different methods for hypothetical available PMPC values. In each table, there are three columns of risk estimates which are, from left to right, based on the regression method, the sample distribution over all system types, and the sample distribution for one system type. These results are derived from two actual cases of data and from the regression model fitted to a subset of the current software maintenance block release data set. Note in table 3-4 that the methods can indeed yield substantially different results.

Table 3-3.

Old and Refined Estimated Risk Methods Yield Similar Results

ESTIMATED ln (PMPC): .9545 SW TYPE: C-E

ESTIMATED PMPC: 2.5974 CASE NO.: 104

AVAILABLE		RISK		
PMPC	ln (PMPC)	REGRESS	ALL SYS.	C-E SYS.
1	0	.8420	.871	.923
2	0.6931	.6083	.639	.723
3	1.0986	.4400	.443	.490
4	1.3863	.3247	.346	.400
5	1.6094	.2454	.257	.284

86-0360-TR-W-07

Table 3-4.

Old [illegible]fined Estimated Risk Methods Yield Different Results

ESTIMATED ln (PMPC) : 2.141 SW TYPE: EW

ESTIMATED PMPC: 8.508 CASE NO.: 136

AVAILABLE		RISK		
PMPC	ln (PMPC)	REGRESS	ALL SYS.	C-E SYS.
1	0	.9879	.871	.773
5	1.6094	.7121	.257	.273
10	2.3026	.4327	.064	.136
15	2.7081	.2753	.014	.045
20	2.9957	.1845	.007	.000

86-0360-TR-W-07a

i. In order to compute the estimated risk using the refined approach, it must be possible to numerically estimate integration over a normal distribution function with a mean and variance. Recall that

$$\hat{L} = b_0 + b_1\,(\text{AVGSKILL}) + b_2\,(\text{PTCORR}) + b_3\,(\text{PCLOW}) + b_4\,(\text{PCHIGH}) + b_5\,(\text{PPNORM}) + \sum_{i=6}^{10} b_i(\text{TYPE}_i)$$

and

$$\hat{\text{PMPC}} = \text{estimated person months per change} = e^{\hat{L}}.$$

Estimated risk for an available PMPC, $PMPC_A$, is calculated as

$$\hat{R} = \text{Estimated Risk} = 1 - F\left(\frac{\ln(PMPC_A) - \hat{L}}{s}\right)$$

where

$F(x)$ = cumulative standard normal distribution function evaluated at x

s = an estimate of the standard deviation of the normal distribution of the L_1 about $\hat{L}$.

The value for "s" is obtained from the BMDP-generated table of regression results (see figure 3-3), where it is labelled "STD. ERROR OF EST." and has a value in this case of 0.9509. The function F may be numerically approximated by

$$F(x) = \frac{1}{2} + \left[SGN(x)\right] \left[\frac{1}{2} G\left(\frac{|x|}{\sqrt{2}}\right)\right]$$

$$G(z) = 1 - \left(1 + a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4\right)^{-4}$$

$a_1 = 0.278393$
$a_2 = 0.230389$
$a_3 = 0.000972$
$a_4 = 0.078108$

$$SGN(x) = \begin{cases} 1 \text{ if } x \geq 0 \\ -1 \text{ if } x < 0 \end{cases}$$

j. As an example, the JTIDS Class 2 Terminal user/supporter baseline estimate (see section 2, figure 2-6) has the following block 3 values for the regression model parameters:

AVGSKILL = 3.0
PTCORR = 0.65
PCLOW = 0.65
PCHIGH = 0.05
PPNORM = 0.65
$TYPE_{C-E}$ = 1.0

Thus,

$$L = 0.07083 + (0.54837)(3.0) + (-0.02948)(.65) + (0.21815)(.65) + (2.09568)(.05) + (-1.57228)(.65) + 0.35255(1)$$

$$= 1.2739275$$

$$\hat{PMPC} = 3.57$$

The available PMPC is computed as

$$\text{PMPC (available)} = [(15*.19 + 9*.90)*(0.667)*(9.0)]/20$$

$$= 3.285$$

The estimated risk is therefore

$$\hat{R} = 1 - F\left(\frac{\ln(3.285) - 1.2739275}{0.9509}\right)$$

$$= 1 - F(-0.0889272)$$

$$= 1 - \left(\frac{1}{2} + SGN(-0.0889272)\left[\frac{1}{2}G\left(\frac{|-0.0889272|}{\sqrt{2}}\right)\right]\right)$$

$$= 1 - \left(\frac{1}{2} + \frac{1}{2}G(0.0628810)\right)$$

$$= 1 - \left(\frac{1}{2} + \frac{1}{2}(1 - (1 + 0.0175056 + 0.0009110 + 0.0000002 + 0.0000012)^{-4})\right)$$

$$= 1 - \left(\frac{1}{2} + \frac{1}{2}(1 - 0.9295990)\right)$$

$$= \frac{1}{2} + \frac{1}{2}(0.0704010)$$

$$= 0.5 + 0.0352005$$

$$= 0.5352005$$

3.7 SOFTWARE SUPPORTABILITY EVALUATION.

a. The RAMSS baseline report (reference 1.4.5) provided an overview of the methodology and an extended example of the methodology application. For the most part the spirit of that information remains current. There are some minor procedural differences in the evaluation process, use of historical maintenance profiles, computation of risk values, and so forth. The major changes in the software supportability evaluation process is in the form and use of the user/supporter baseline estimate, and in the improvements of software life cycle process evaluation.

b. The user/supporter baseline estimate is recommended for use in all three software supportability evaluations: software life cycle process, software product maintainability, and software support resources. However, its use is probably more important as a mechanism to stimulate using and supporting command discussion regarding the projected personnel resources and change profile workload. The baseline estimate would be more useful in this sense during the evaluation calibration activity than through direct reference during the actual evaluations. The RAMSS still uses the concept of evaluating against the baseline, so it is understood what risk means (i.e., a negative event means the baseline workload estimate was not met).

c. The software life cycle characteristics in reference 1.4.5 inadvertently did not include the implementation methods. Project cost has been changed to organization structure, and organization interfaces has been changed to project interfaces. This new terminology is consistent with the data collected during the generation of historical maintenance profiles. In addition, the configuration management system characteristic in the software support resources/support systems hierarchy was unintentionally not included. These minor changes are more editorial in nature, but can create confusion if referencing across prior documents.

d. For purposes of consistency, the following reports should be considered important references for the stated reasons. All refinements of the RAMSS are in the new baseline RAMSS documents.

(1) <u>Old RAMSS Baseline Report (reference 1.4.5)</u>. Use this report for background into the process through which the RAMSS has been derived. All historical data, procedures, and methodology are more accurately reflected in the new baseline documents.

(2) <u>RAMSS User's Handbook (reference 1.4.6)</u>. This report is one of the new baseline RAMSS documents. Use it to understand how to implement a RAMSS, obtain evaluation analysis reports, and update/report contents of the historical data bases through the automated support system for RAMSS.

(3) <u>RAMSS Adaptation Guidelines (reference 1.4.7)</u>. This report is one of the new baseline RAMSS documents. Use it to understand how to adapt the current AFOTEC software supportability evaluations to the requirements of the RAMSS. In addition, an appendix contains the Software Life Cycle Process Evaluator's Guide.

(4) <u>RAMSS Pilot Study and Methodology Refinements</u>. This report is one of the new baseline RAMSS documents. It includes results of a pilot study using RAMSS, an RAMSS Evaluator's Guide (appendix B), and a briefing of RAMSS (appendix A) for general use in describing the main features of the RAMSS.

(5) <u>AFOTECP 800-2 Series</u>. This AFOTEC pamphlet series is basic to the RAMSS since the Software Product Maintainability (volume 3) and the Software Support Resources

(volume 5) Evaluator's Guides are included. Updates to the pamphlet series, including Management of Software Operational Test and Evaluation (volume 1), should carefully consider implications of the new baseline RAMSS documents. In particular, the RAMSS Evaluator's Guide and Software Life Cycle Process Evaluator's Guide were written in a form which should be easily adaptable as AFOTECP 800-2 volumes.

IV. Conclusions and Recommendations

SECTION IV

CONCLUSIONS AND RECOMMENDATIONS

4.1 INTRODUCTION.

This section summarizes some of the more important issues which will affect the production use of the RAMSS by AFOTEC personnel. It must be understood that any risk assessment will only be as good as the data upon which it is based, and the process through which the data are analyzed and conclusions derived. Most of the conclusions and recommendations of this report are in some way concerned with the data collection or the implemented process necessary to obtain a valid risk assessment of software supportability.

4.2 USING RAMSS FOR AFOTEC PROGRAMS.

The following conclusions/recommendations have been derived from the process of developing the RAMSS.

(1) The RAMSS is not mature, but it should provide useful analysis results and conclusions. The RAMSS must evolve through application in order to reach its full potential. The evolution includes updating the historical evaluation and maintenance release data and the associated risk regression equations, and refining the procedural aspects of applying the RAMSS to actual software assessments.

(2) The User/Supporter Baseline Estimate is a useful mechanism to facilitate using command, supporting command, and AFOTEC personnel interaction concerning computer resource support requirements. This Estimate is valuable input to the calibration briefing/discussion prior to software supportability evaluations. The Estimate has limited use during completion of software supportability evaluation

questionnaires. The Estimate can be derived by AFOTEC personnel using the RAMSS automated support system independent of using/supporting command personnel for use in early life cycle high level software supportability evaluations.

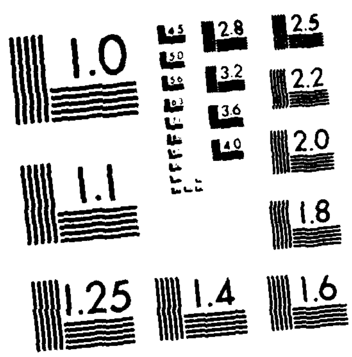
(3) The update of the regression equations to reflect new evaluation and/or maintenance release data should be carefully controlled. One or two updates a year should be sufficient. Each update should include a thorough statistical analysis of the BMDP statistical reports. Instructions for performing the updates are contained in the RAMSS User's Handbook (reference 1.4.6).

(4) AFOTEC should maintain a lessons learned history of the Software Life Cycle Process evaluations so the procedures for collecting information and mapping the information into the appropriate questionnaire responses can become more consistent and routine.

(5) The RAMSS historical data base is partly subjective. Continued data collection should provide more accuracy and maintain the currency of the information.

(6) It is strongly recommended that a position of RAMSS system manager be created and filled by an AFOTEC person (e.g., a civilian) who would provide continuity from program to program and year to year. The RAMSS system manager would provide expertise to the STM/DSE for each software OT&E effort, maintain the RAMSS and supportability evaluation procedures, and operate the automated support analysis tools (e.g., RAMSS, QAP, and ASSET). Perhaps the most important concern in use of the RAMSS by AFOTEC is its consistent application. With the nature of AFOTEC personnel's temporary duty assignments, [illegible]

AD-A191 875 RISK ASSESSMENT METHODOLOGY FOR SOFTWARE SUPPORTABILITY 2/3
(RAMSS): PILOT EV..(U) BDM CORP ALBUQUERQUE NM
D E PEERCY ET AL. 14 APR 86 BDM/ABQ-86-0360-TR
UNCLASSIFIED F29601-85-C-0050 F/G 12/5 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

unlikely that a corporate knowledge of RAMSS and the software supportability evaluations can be maintained without positive action to create a permanent RAMSS system manager position.

(7) The major documents for understanding and using the RAMSS are:

a) Software Supportability Risk Assessment in OT&E: Historical Baselines Risk Profiles, BDM/A-85-0510-TR, Volumes I and II, October 7, 1985

b) Risk Assessment Methodology for Software Supportability (RAMSS): User's Handbook, BDM/A-85-1270-TR, April 14, 1986

c) Risk Assessment Methodology for Software Supportability (RAMSS): Guidelines for Adapting Software Supportability Evaluations, BDM/ABQ-86-0090-TR, April 14, 1986

d) Software Life Cycle Process Evaluator's Guide, BDM/ABQ-86-0090-TR, Appendix A, April 14, 1986

e) Risk Assessment for Software Supportability (RAMSS): Pilot Evaluation Results and Methodology Refinement, BDM/ABQ-86-0360-TR, April 14, 1986.

f) Overview Briefing of RAMSS, BDM/ABQ-86-0360-TR, Appendix A, April 14, 1986

g) Risk Assessment Methodology for Software Supportability (RAMSS): Evaluator's Guide, BDM/ABQ-86-0360-TR, Appendix B, April 14, 1986

h) AFOTECP 800-2, Software OT&E Guidelines, Volume III, Software Maintainability Evaluators Guide, Volume V, Software Support Resource Evaluation User's Guide.

The Software Life Cycle Process Evaluator's Guide and the RAMSS Evaluator's Guide should be adapted as part of the AFOTECP 800-2 series. Volume V of the AFOTECP 800-2 series is no longer being published and should be appropriately updated to make the use of the ASSET automated support tool more effective.

4.3 PILOT EVALUATION

The following list summarizes the conclusions/recommendations from the pilot study application of RAMSS to the software.

(1) The user/supporter baseline estimate (USBE) was able to be derived, but required some reasonable "guesses" based upon maintenance release data.

(2) The main benefit of the USBE was the interaction and discussion among using command, supporting command, and AFOTEC personnel.

(3) The USBE was not a major factor in answering the individual evaluation questions.

(4) The Software Life Cycle Process (SLCP) evaluation cannot be done in the same manner as the other evaluations. For credibility, it is essential to capture the life cycle process characteristics over time to create a "history" base upon which responses to the SLCP questions can be based.

(5) The use of the RAMSS tool will aid the interpretation of the risk assessment results, but there are several more "what if" analysis functions that could be developed.

(6) One important side effect of the USBE evolution process is the using and supporting command personnel identification of areas of risk. These areas of risk can be investigated by AFOTEC personnel for potential impact upon the system OT&E as well as the software portion of the OT&E.

(7) The using and supporting command personnel were very cooperative during the pilot study. They seemed to appreciate the opportunity to participate in the specification of the USBE.

(8) The development effort has several "generic" life cycle process flaws which have been observed across many systems:

a) The full-scale development schedule of 27 months defined in 1980 was much too ambitious. Current projections are for PMRT in 1990.

b) Functional expectations changed from the prototype demonstration.

c) Interoperability requirements with other services were (are) a source of problems.

d) Planning for computer support resources during the post-deployment phase has been very poor. Generally, very little priority is given to this function.

e) Organizational centralization of responsibility and consistency of personnel over the project life cycle has been poorly managed.

f) Configuration management plans, procedures, and automated tool support are inadequate.

g) Security concerns (e.g., classified software documentation and source listings) have not been properly addressed.

h) Procurement activity understanding of the deliverable requirements, as reflected in the RFP/CDRL/etc., is inadequate in the area of computer resources, test support, and quality assurance.

(9) There is a significant amount of project management and configuration management which is being done, but not being properly incorporated into the proper planning, specifications, and other documents. For example, the JTIDS Class 2 Terminal support personnel knew much more information concerning the plans, organization structure, test strategies, configuration control, personnel allocations, and facility layout than was contained in the TEMP, CRISP, or O/S CMP.

4.4 DERIVING A USER/SUPPORTER BASELINE ESTIMATE.

a. The theoretical basis of the RAMSS requires the use of a user/supporter baseline estimate of support resources and workload change profile in order to have a baseline against which a software supportability evaluation can be conducted. Thus, the measure of risk derived from the evaluation scores is relative to meeting the baseline workload estimate. Without such an estimate, the evaluation would not have a specified baseline and it would be more difficult to interpret the resulting derived risk (i.e., what would constitute a negative event?).

b. The requirement that using and supporting command personnel arrive at a consensus on the baseline estimate is not essential. It is not even necessary that the using and supporting command participate in the derivation of the baseline estimate. The baseline estimate could be derived by AFOTEC personnel using the historical maintenance release data, computer resources support planning documents, and the RAMSS automated support system. The resulting baseline estimate and subsequent software supportability risk assessment would be consistent and could be appropriately reported by AFOTEC.

c. Although the RAMSS does not require using and supporting command personnel participation, it is highly recommended. The benefits of this participation during the Pilot Evaluation were significant. The communication among using command, supporting command, and AFOTEC personnel significantly improved the accuracy of the baseline estimate. The understanding of follow-on support requirements among the participants was greatly improved. Areas of supportability risk were identified by both using and supporting command personnel. Results of the discussions should aid in future updates to the CRISP and O/S CMP.

d. The using and supporting command personnel were very supportive, and seemed pleased to be involved in the process of deriving a baseline workload estimate.

e. The user/supporter baseline estimate derivation process consists of four basic steps any of which may serve as the starting point, and all of which may require some iteration.

(1) STEP 1: Derive a draft user/supporter baseline estimate from the RAMSS automated support tool entry and analysis procedures.

(2) STEP 2: Obtain comments from the supporting command personnel on the draft user/supporter baseline estimate.

(3) STEP 3: Obtain comments from the using command personnel on the draft user/supporter baseline estimate.

(4) STEP 4: Derive a compromise from the draft estimate and using/supporting command comments.

The contact with the using/supporting command personnel can be through on-site visits and/or telephone conversations.

f. The user/supporter baseline estimate should be discussed prior/during the software supportability evaluations. The most likely focus is during the evaluator calibrations for the software product and software support resources evaluations. The software life cycle process evaluation is a more long-range process in which early data collection will provide information for the baseline estimate.

g. There was very little opportunity to determine the effect of not using a baseline estimate. The possibility of not having using and supporting command personnel participation has been considered above. An evaluation could be performed with no baseline estimate, but very little additional information above the evaluations scores could be obtained. In particular, there could be no estimated risk computation and the evaluated risk would have no baseline interpretation upon which to interpret the meaning of the risk value. Since AFOTEC personnel can derive a baseline estimate independent of other participants, there would seem to be no good reason why a baseline estimate could not be derived.

4.5 DATA COLLECTION TO UPGRADE RAMSS.

a. The data from future AFOTEC software supportability evaluations should be entered into the historical evaluation data base. Maintaining these data from actual evaluations is critical to the evolution of the RAMSS.

b. It is critical that maintenance release data continue to be collected and entered into the historical maintenance release data base. These data are the basis for connecting actual maintenance activity with the AFOTEC software supportability evaluation results. In order for support site personnel to obtain the necessary release data, it is necessary to make the data collection process efficient and somewhat related to activity already being accomplished. A recommended data collection form and procedure is discussed in references 1.4.5 and 1.4.6. The essential elements of the the form and procedure are:

(1) The form and procedure are temporary until a more permanent arrangement can be integrated into the Air Force software support concept.

(2) All cognizant software support sites and major (critical) software systems currently being supported should be solicited to participate in the data collection effort. Initially it is recommended that AFOTEC contact personnel responsible for the systems currently in the historical data base and request continued support for the collection of maintenance release data.

(3) It is estimated that completion of the data collection form (and altering current practices so the data are readily available) would take very little additional support personnel time. The range might be from one person day to one person week per release.

(4) The data collection form data elements required for each release include: site, system, software system, software system type, size in thousands of source lines, source languages, personnel counts and skill levels, release identification, release start date, engineering completion date, field release date, and baseline software support profile data on each change request in the release.

(5) The data collection procedure would involve each support site completing a data collection form for each software system release. The form would be sent to a data repository site (AFOTEC, at this time) for integration into the current data base, update of the historical maintenance profiles, and further statistical analysis.

(6) It is recommended that such a data collection form be adopted and that AFOTEC develop the necessary data base and analysis environment to support regular revisions to the historical maintenance profiles. The current RAMSS automated support system, (see RAMSS User's Handbook, reference 1.4.6) is adequate to accomplish this function.

4.6 SUMMARY.

a. In summary, the RAMSS should be an effective tool for AFOTEC personnel to use in assessing the risk to the Air Force of being able to provide adequate support for mission-critical software.

b. It is important for AFOTEC personnel to understand and properly apply the RAMSS for best results. Because of the natural transition of personnel it is difficult for AFOTEC to maintain corporate knowledge. It is strongly recommended that a RAMSS system manager position be created and filled by a person who can provide long-term continuity for the methodology, automated tool support, and implementation of software supportability evaluation guidelines.

c. The data collection for evaluation data and maintenance release data should be continued. New data should be entered into the RAMSS historical data bases and the resulting RAMSS software supportability risk regression equations updated. The RAMSS can only be as effective as the accuracy and currency of its baseline historical data.

Appendix A
Overview Briefing of RAMSS

APPENDIX A

OVERVIEW BRIEFING OF RAMSS

The purpose of this appendix is to provide AFOTEC with a set of materials which can be used to brief the background, purpose and procedures of the risk assessment methodology for software supportability (RAMSS). The materials are presented in a storyboard fashion to permit a briefer to understand the information contained in each slide. The materials presented here are probably not suited to every situation. Therefore, the briefer may need to tailor the materials to varying purposes and audiences. In any event, the materials contained in the appendix will provide a place to start when requirements for general information on the RAMSS exist.

OVERVIEW OF THE RISK ASSESSMENT METHODOLOGY FOR SOFTWARE SUPPORTABILITY (RAMSS)

(Date)

(Presenter)

(Oganization)

(Audience)

The purpose of this briefing is to give you an overview of the Risk Assessment Methodology for Software Supportability (RAMSS). The methodology was developed by The BDM Corporation for AFOTEC to provide the Air Force with a structured method for determining the supportability of software acquired for mission critical systems. Application of this methodology will provide software testers with areas which require testing emphasis, and decision makers with an assessment of the software supportability risk. Significant benefits result from the capability to measure the software's supportability risk, compare this risk with historical data, pinpoint software supportability risk drivers, and project alternative choices for risk reduction.

OVERVIEW

- AFOTEC ORGANIZATION
- BACKGROUND
- METHODOLOGLY
- AN EXAMPLE
- CONCLUSIONS

The briefing will consist of five major sections. The first section will be a review (optional) of AFOTEC--its purposes, structure, and activities. The second section will be a brief review of the development background for the risk assessment methodology. This review will be followed by a discussion of the methodology, and an application of the methodology via a sample problem. Concluding remarks will finish the briefing.

REFERENCES

Peercy, D., M. Estill, W. Huebner, "Software Supportability Risk Assessment in OT&E: Historical Baselines for Risk Profiles," Volumes I and II, BDM/A-85-0510-TR, October 7, 1985.

Peercy, D., M. Estill, W. Huebner, K. Shaw, "Risk Assessment Methodology for Software Supportability (RAMSS) User's Handbook," BDM/ABQ-85-1270-TR, April 14, 1986.

Peercy, D., W. Huebner, "Risk Assessment Methodology for Software Supportability (RAMSS): Guidelines for Adapting Software Supportability Evaluations," BDM/ABQ-86-0090-TR, April 14, 1986.

Peercy, D., W. Huebner, M. Estill, K. Shaw, "Risk Assessment Methodology for Software Supportability (RAMSS): Pilot Evaluation Results and Methodology Refinement," BDM/ABQ-86-0360-TR, April 14, 1986.

AFOTECP 800-2, Volumes I through V, Software OT&E Guidelines. (Volume V is no longer being published.)

The major references for the information contained in the briefing are shown. The AFOTEC series of pamphlets documents the tools which were incorporated into the risk assessment methodology and which were the primary means of assessing software supportability prior to the development of RAMSS. The other BDM documents contain all the information necessary to plan, conduct, analyze, and report the risk assessment of software supportability.

DEFINITIONS

SOFTWARE SUPPORTABILITY: A MEASURE OF THE ADEQUACY OF PRODUCTS, RESOURCES, AND PROCEDURES TO FACILITATE:

(1) INSTALLING AND MODIFYING SOFTWARE

(2) ESTABLISHING ON OPERATIONAL BASELINE

(3) MEETING USER REQUIREMENTS

SOFTWARE SUPPORTABILITY RISK: THE PROBABILITY AT A GIVEN POINT DURING THE SOFTWARE SYSTEM LIFE CYCLE THAT THE PREDICTED SOFTWARE MAINTENANCE ACTIVITIES CAN NOT BE ACCOMPLISHED WITH THE AVAILABLE (OR PLANNED) SOFTWARE SUPPORT RESOURCES.

Before going further with this briefing, it is essential that a common understanding of the significant terms be obtained. Software supportability encompasses all of the personnel, resources, *and procedures required to perform change activity on the software. This change activity includes* modifying (correcting, enhancing, or converting) and installing software which forms an operational software *baseline meeting the requirements of the user. The measure of* software supportability in terms of risk contains several important ideas. First, the risk should be computable at any point during the software system's life cycle. Second, to compute the risk, an estimate of the software maintenance activities (in the form of anticipated personnel, procedures, and number of software change requests) must be obtained. And third, the risk computed is a measure of the inability of the available (or planned) software support resources to meet the predicted software maintenance requirements.

SOFTWARE SUPPORTABILITY PROBLEM

1. SUPPORT FOR MAJOR WEAPON SYSTEMS IS A MAJOR COST FACTOR.
2. SOFTWARE IN INCREASINGLY COMPLEX EMBEDDED COMPUTER SYSTEMS IS BECOMING MORE CRITICAL TO MISSION EFFECTIVENESS.
3. SOFTWARE SUPPORTABILITY IS NOT EASILY MEASURED.

If the issue of software supportability, as previously defined, were not a problem, then there would be little need for the development of a risk assessment methodology. However, software supportability is a problem for concern because the overall supportability for major weapon systems is a significant cost factor. As these weapon systems become increasingly sophisticated, complex embedded computer systems are becoming more required and critical to meet the needs of the user, and hence to overall mission effectiveness. The most significant issue, however, is that the software support activities, and all the events which occur with a software product prior to its delivery, involve complex processes which are not easily measured. There are so many factors which can have a negative impact on the quality of the final software product, that a structured approach to identify and account for those factors is required in order to begin to adequately address the problems. The next slide discusses these issues in a little more detail.

SOFTWARE SUPPORTABILITY PROBLEM
(Continued)

ISSUES:

1. HOW DO YOU DETERMINE "SUPPORT" REQUIREMENTS FOR SOFTWARE PRODUCTS? (FACILITIES, PEOPLE, PROCEDURES)
2. WHAT FACTORS SHOULD BE MEASURED?
3. HOW SHOULD THE FACTORS BE MEASURED?
4. HOW SHOULD SUPPORT DECISION OPTIONS BE PRESENTED TO THE DECISION-MAKER?

There are several issues which complicate the software supportability problem. The first issue concerns the methods that are used to determine what facilities, personnel, and procedures will be necessary to support software once it is delivered. The fact is, there are no standard methods for determining support requirements. Another important related issue is that support requirements are continually evolving throughout the software life cycle process, which further complicates things. Embedded in each of these requirements are various factors which can influence the software supportability. Such situations as having the proper skill levels available for maintenance personnel, having the right computer equipment installed, having appropriate refrigeration for the computers, having documentation available for all software, having a good software configuration management system, and so on, can have a severe impact on the supporter's ability to maintain the system. Assuming that all the factors could be identified, the next issue is, how are those factors to be measured? How do you know if the software documentation is of high quality, or if there is enough floor space to house all the support equipment, or if the software was designed to permit easy modification? Finally, assuming that all the factors which influence the software support activity could be measured, how do you convey this information in a form or format which adequately describes the ability to support the software to a high-level decision maker, and presents options for solving noted deficiencies. The risk assessment method presented in this briefing does not solve all of these issues precisely, but all of the issues have been considered and addressed and a structured solution to each problem is presented.

WHAT DOES RAMSS PROVIDE AFOTEC?

A Structured Tool for Measuring the Risk (Probability) That Support Resources Cannot Adequately Support Delivered Software

The Tool Measures Risk:

- **At Any Point During the System Development Life Cycle (Hence in Enough Time to Identify and Correct Deficiencies)**
- **At a Level Which Identifies Risk Drivers**
- **With Confidence That the Risk is a Real Value Based on Historical Activity**
- **Using a Minimum Amount of AFOTEC/Government Personnel and Equipment Resources**
- **In a Manner Which Provides Results Easily Explained to Software Testers and Decision Makers**

The primary benefit to AFOTEC from using the RAMSS is a capability to measure the probability, based on historical information, that current (or planned) support resources cannot adequately support delivered software. The methodology attempts to minimize the subjectivity involved in current software supportability evaluation methods. The methodology has several significant advantages for AFOTEC:

(1) It can be applied at any time during the system's development life cycle (including very early in the cycle)

(2) It identifies those characteristics which are driving the supportability risk

(3) It provides a real risk value which is based confidently on historical support activity data

(4) It is a comprehensive method requiring minimum time and resources to perform

(5) It provides tradeoff analysis results easily understood by software testers and decision makers.

AFOTEC SUBTASKS TO DEVELOP RAMSS

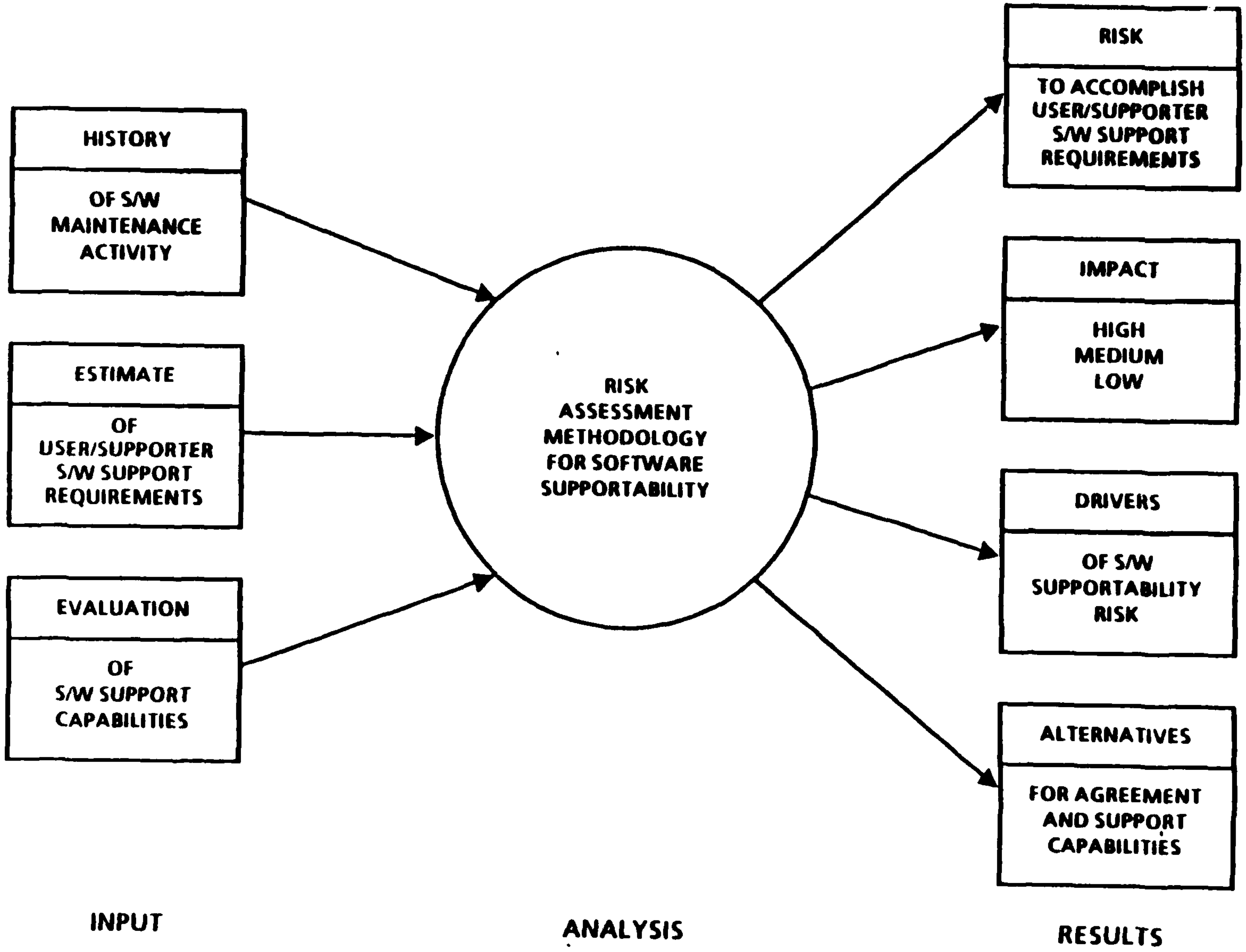
- SS/304 (APRIL 1984 - OCTOBER 1984)
  - FEASIBILITY STUDY
  - LITERATURE REVIEW (6400 DOCUMENTS)
  - INITIAL FRAMEWORK IDENTIFIED
- SS/327 (NOVEMBER 1984 - JULY 1985)
  - COLLECT SITE SUPPORT DATA (SIX SITES, 81 SYSTEMS, 336 BLOCK RELEASES)
  - REFINE FRAMEWORK
  - ANALYZE RESULTS (REGRESSION, FACTOR ANALYSIS)
- SS/412 (OCTOBER 1985 - MARCH 1986)
  - PILOT STUDY (JTIDS CLASS 2 TERMINAL)
  - REFINE FRAMEWORK
  - PROVIDE TOOL FOR AFOTEC PRODUCTION USE

The efforts to develop a risk assessment methodology for software supportability began in April of 1984 with a feasibility study which consisted of a literature review, an analysis of the review results, and a recommendation to AFOTEC on the feasibility of developing a methodology for risk assessment. The basic results of the literature review and analysis were that no current method existed which satisfied AFOTEC's requirements; however, some parts of existing methodologies were adaptable. This initial effort concluded by identifying a framework for assessing software supportability risk. This framework adapted previous methods and combined the risk assessment ideas with a need to measure risk against historical software maintenance data. The recommendation of this study was that it appeared feasible to develop a risk assessment methodology, and AFOTEC agreed.

Given the conclusion of the first study, the next effort was to collect historical software maintenance data which could serve as a baseline against which to assess the risk. This data consisted of specific information on weapon systems by type, the number and types of software changes being performed for each block release (baseline) of the software, an estimate of the resources expended in performing software changes, and a subjective risk assessment of various characteristics relating to software maintainability. Based upon the data obtained, it was possible to build historical maintenance profiles against which the supportability risk of future systems could be assessed. As a result of the data collected, the RAMSS was modified to accommodate the data available, rather than the detailed data desired. All data were subjected to extensive statistical analysis to verify validity and accuracy. Regression and factor analysis were performed on the subjective risk assessment data to verify that the characteristics being addressed by the RAMSS were in fact representative of the factors which the support community recognized as being important.

The last step in the RAMSS development was a pilot study for an ongoing AFOTEC program (the JTIDS Class 2 Terminal was chosen by AFOTEC). This pilot study exercised all aspects of the risk assessment process, and resulted in minor modifications to the RAMSS framework. In addition, the effort defined a high-level method for evaluating the software life cycle process, and developed a set of software tools for maintaining the historical data base of software maintenance data and computing the RAMSS output products.

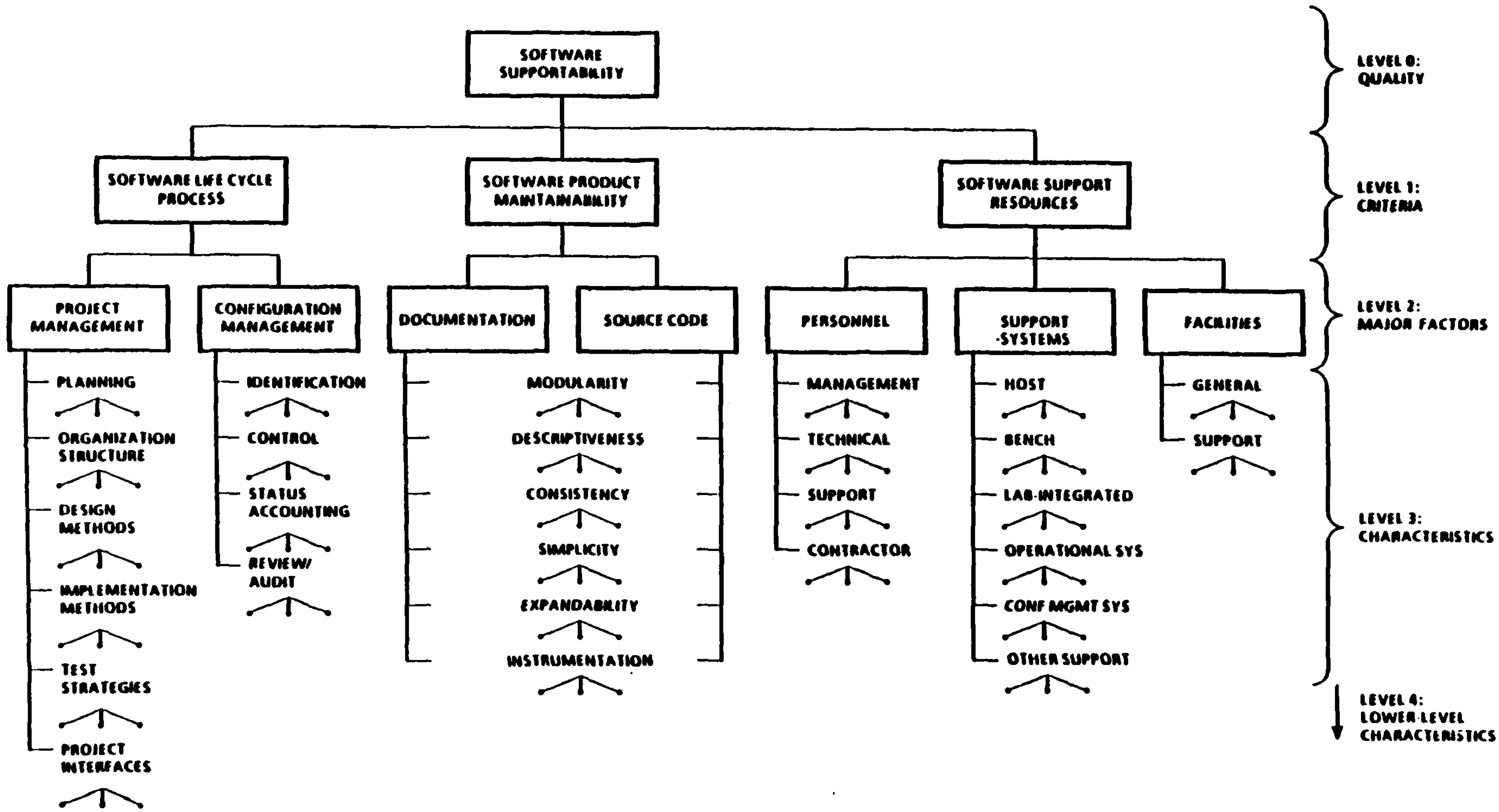
HIGH LEVEL VIEW OF RAMSS



This figure shows the interface of external information with the RAMSS. There are three basic elements which are required to perform the risk assessment methodology: 1) the historical maintenance activity data; 2) an estimate, obtained by iterative dialogue with the using and support agencies, of the software support requirements for the system being assessed; and 3) an evaluation of the system being assessed, using the software supportability evaluation tools discussed on the next slide.

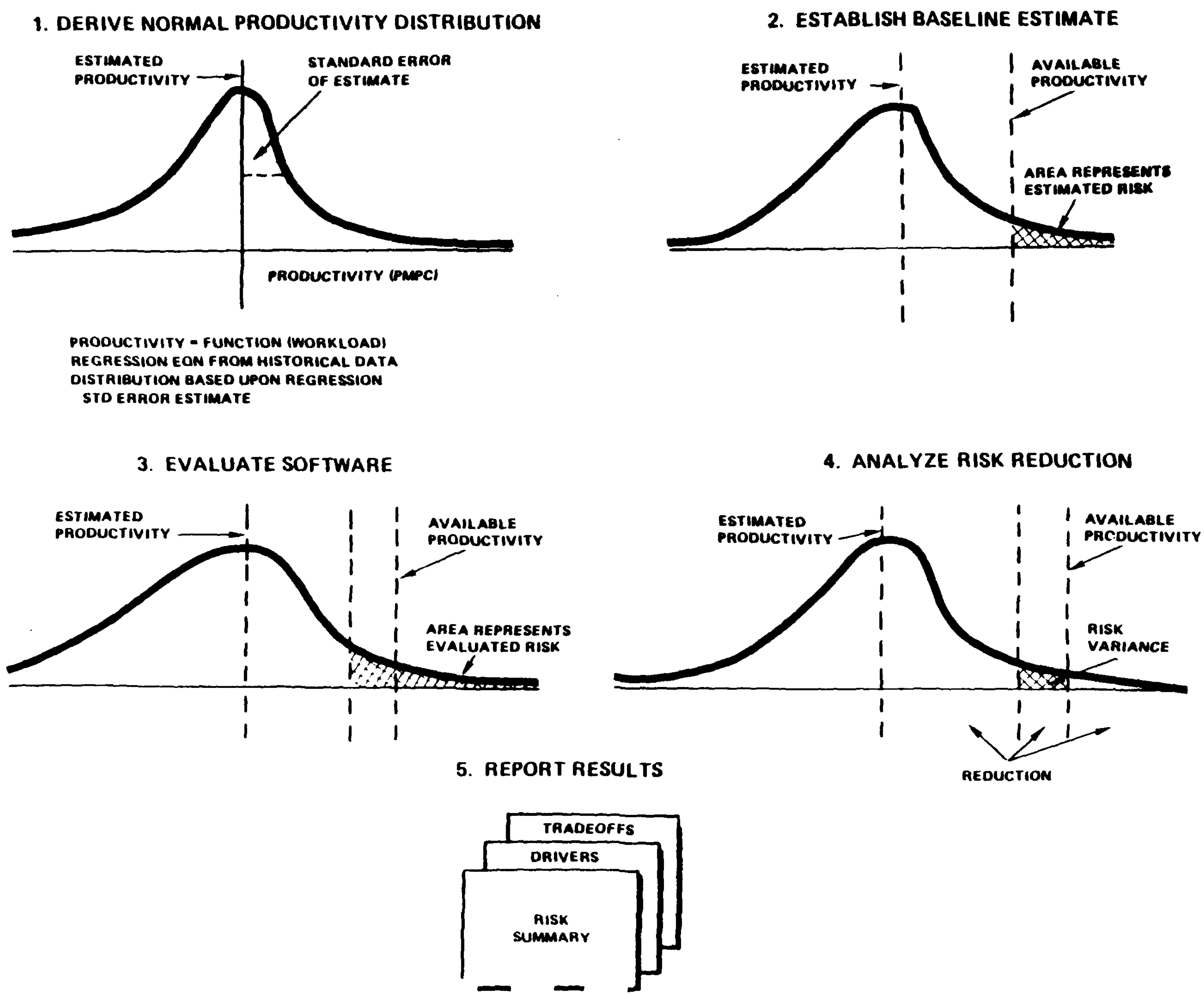
These elements are used by the RAMSS, in a manner to be described shortly, to produce the information shown as results. The results demonstrate the four basic outputs, and also advantages of the RAMSS. First, the risk to accomplish user/supporter software support requirements is presented as a probability of not being able to support those requirements. Second, partly because of the ability to compare estimated and evaluated risk with historical data, the impact may be classified as high, medium, or low. Third, the hierarchical structure of the evaluation process enables a quick determination of the risk drivers. And fourth, because of the structural approach to risk determination, it is easier to perform alternative trade-off analyses to examine viable methods of decreasing the risk.

EVALUATION HIERARCHY



This figure illustrates the hierarchical approach used by the RAMSS to evaluate the software supportability of a system. By dividing the evaluation into three criteria (software life cycle process, software product, and software support resources) and examining the major factors and characteristics of each of these criteria, it is easier to identify risk drivers. Of particular interest are those characteristics which have poor ratings as a result of applying the evaluation. The evaluations are conducted by asking a team of evaluators to rate low-level characteristics on a scale of 1 (worst) to 6 (best) in response to a statement about each characteristic's contribution to the overall maintainability of the software system. For example, evaluators may be asked to rate whether they strongly disagree (giving a rating of 1) or strongly agree (giving a rating of 6) that the "comments in this module contain useful information." The evaluators respond to a series of subjective and objective statements about each characteristic. The responses are analyzed and combined to form an overall score for each of the criteria (software life cycle process, software product, and software support resources), which in turn are combined to form an overall score for software supportability. The risk assessment methodology converts the score to a risk number, as discussed in the next slide.

Each of the three elements discussed above is a separate evaluation tool. The documentation for the software life cycle process evaluation may be found in BDM/ABQ-86-0090-TR. The documentation for the software product evaluation may be found in AFOTECP 800-2, Volume III. The documentation for the software support resources evaluation (also known as ASSET) may be found in AFOTECP 800-2, Volume V.


1. DERIVE NORMAL PRODUCTIVITY DISTRIBUTION
ESTIMATED PRODUCTIVITY
STANDARD ERROR OF ESTIMATE
PRODUCTIVITY (PMPC)
PRODUCTIVITY = FUNCTION (WORKLOAD)
REGRESSION EQN FROM HISTORICAL DATA
DISTRIBUTION BASED UPON REGRESSION
STD ERROR ESTIMATE
2. ESTABLISH BASELINE ESTIMATE
ESTIMATED PRODUCTIVITY
AVAILABLE PRODUCTIVITY
AREA REPRESENTS ESTIMATED RISK
3. EVALUATE SOFTWARE
ESTIMATED PRODUCTIVITY
AVAILABLE PRODUCTIVITY
AREA REPRESENTS EVALUATED RISK
4. ANALYZE RISK REDUCTION
ESTIMATED PRODUCTIVITY
AVAILABLE PRODUCTIVITY
RISK VARIANCE
REDUCTION
5. REPORT RESULTS
TRADEOFFS
DRIVERS
RISK SUMMARY

There are five basic steps to the risk assessment methodology.

Step 1a: Collect Historical Data on Software Maintenance Activities. The result of this effort is a regression equation which predicts productivity in person months per change as a function of workload parameters: average skill, type of software system (e.g. EW, OFP), and profile of software change requests per block release. The change profile includes number, type, complexity, and priority of the changes.

Step 1b: Collect Historical Data of Software Supportability Evaluation. The result of this effort is a regression equation which predicts the risk of accomplishing a baseline of software maintenance activity based upon the software product maintainability, software life cycle process, and software support resources. This evaluated risk is related to the estimated risk through the baseline.

Step 2: Establish a Baseline Estimate for the Current System's Estimated Maintenance Requirements. This estimate includes the available productivity and expected workload information for up to three block releases. This information is input to the historical productivity regression equation to derive the estimated productivity in person months to accomplish the expected workload, and the estimated risk that the available productivity will not be sufficient to accomplish the expected workload. The normal distribution curve illustrates the estimated risk as the shaded area under the curve.

Step 3: Perform Software Supportability Evaluation. The evaluation is conducted against the Baseline Estimate and the evaluation data is input to the evaluated risk regression equation. The evaluated risk is computed and represents the risk due to the evaluated factors of not being able to accomplish the expected workload with the available productivity. This evaluated risk also represents an area under the normal distribution curve. The "difference" in the estimated risk and the evaluated risk represents the amount of risk reduction (if any) which probably should be accomplished, assuming the original estimated risk is reasonably acceptable.

Step 4: Analyze Methods to Reduce Risk. There are several possibilities for reduction of both estimated and evaluated risk. The Baseline Estimate available productivity and expected workload parameters can be modified. This will affect both estimated and evaluated risk. The evaluated risk major factor drivers can be reviewed for possible improvement. The amount of improvement possible will be the constraint on the amount of reduction in the evaluated risk.

Step 5: Report Results. The analyses performed in step 4 are summarized and condensed to an analysis report for software testers and a final report for decision makers.

CASE STUDY EXAMPLE
(HYPOTHETICAL)

- BASELINE ESTIMATE
- QUESTIONS/OBJECTIVES
- ESTIMATED SUPPORTABILITY RISK
- EVALUATED SUPPORTABILITY RISK
- ESTIMATED AND EVALUATED RISK INTEGRATION
- TRADEOFF ANALYSIS

The following example will provide additional information on the RAMSS procedure. This example is hypothetical and is meant to be representative of the process, not of any particular system or software support facility. The example will discuss the structure of the baseline estimate and objectives of the RAMSS technical approach. More information will also be presented on the computation of the estimated and evaluated supportability risk. Finally, we will discuss the integration of the risks and potential trade-off analysis.

CASE STUDY EXAMPLE (HYPOTHETICAL) BASELINE ESTIMATE

- **SYSTEM PROFILE**

SYSTEM:	F-16
SWSYSTEM:	COMNET TERMINAL
SWTYPE:	C-E
SUPPORTER:	WR-ALC
USER:	HQ-TAC

- **SUPPORT CONCEPT**

RELEASE SCHEDULE:	9 MONTH BLOCK RELEASE WITH 3 MONTH OVERLAP
SUPPORT STAFF:	15 PERSONS, 19% DEDICATED, AVG SKILL 3.0
	9 PERSONS, 90% DEDICATED, AVG SKILL 3.0

- **BASELINE SUPPORT PROFILE**

BLOCK	TOTAL # CHANGES	TYPE (C, H, V)	COMPLEXITY (H, M, L)	PRIORITY (E,U,N).
1	15	(15, 0, 0)	(0, 0, 15)	(0, 0, 15)
2	20	(15, 5, 0)	(1, 4, 15)	(1, 4, 15)
3	20	(13, 7, 0)	(1, 6, 13)	(1, 6, 13)

The above hypothetical example shows the type of software maintenance activity data which is required to establish a user/supporter baseline estimate. Once finalized, this data becomes the "estimate" of support activity requirements. There are three elements to the estimate: 1) system profile, 2) support concept, and 3) baseline support profile. The system profile is simply identification data for the system of interest. The support concept identifies the planned release schedule for each block release, the projected support staff (to include shared resources), and the support personnel average skill level (1-low to 5-high). The baseline support profile identifies the total number of software changes expected in each block release. These changes are grouped into the major categories of type (correction, enhancement, conversion), complexity (high, medium, low), and priority (emergency, urgent, normal). Definitions of these terms are found in the references. The numbers used for this example are representative of the distribution that might normally be expected from the weapon systems for which historical data was collected. The baseline estimate is obtained by iterative dialogue among AFOTEC and the system's using and supporting commands.

CASE STUDY EXAMPLE
(HYPOTHETICAL)

QUESTIONS ADDRESSED BY RAMSS

- What is the Estimated Risk for the Baseline?
  - Computed From Regression Equation Which Estimates the Person Months per Change From Regression Parameters: Average Skill, Block Release Change Profile and Software Type
- What is the Evaluated Risk for the Baseline Block 2?
  - Supportability Metrics from AFOTEC Evaluation
  - Computed Using Regression Equation of Supportability Metrics
- What Options for Trade-Offs Exist to Reduce the Risk?
  - Baseline Estimate
  - Supportability Metric Changes
  - Historical Baseline Variance/Confidence Range

The basic questions which the RAMSS addresses include computing the "estimated" and the "evaluated" risk. The "estimated" risk, as shown in the following example, is computed from a regression equation which uses the baseline estimate information shown in the previous slide. The "evaluated" risk is computed using the AFOTEC evaluation tools' scores and a regression equation of software supportability metrics. Tradeoff options (as shown) are possible in several different areas. Specifics of these methods will be discussed in the following slides.

CASE STUDY EXAMPLE
(HYPOTHETICAL)

ESTIMATED SUPPORTABILITY RISK

- Available Productivity (Person Months Per Change)
  - Full Time Equivalent * Release Cycle Length /Number Changes
- Estimate Productivity (Person Months Per Change)
  - "Optimum" Level of Effort
  - Derived from Regression Equations

$$PMPC = EXP(L^{\wedge})$$

$$L^{\wedge} = B_o + B_1(AVGSKILL) + B_2(PTCORR) + B_3(PCLOW) + B_4(PCHIGH) + B_5(PPNORM) + \sum_{i=6}^{10} B_i(TYPE_i)$$

- Estimated Risk
  - Derived from Area Under Normal Curve
  - Normal Curve Has Mean (Estimated Productivity) and Variance (Standard Error of Estimate) from the Regression Equation

This slide illustrates the process of computing the estimated risk from the baseline estimate data. The available person months per change is computed as the full time equivalent personnel over the length of the block release cycle divided by the total number of changes in the block release. The estimated person months per change is computed from a regression equation using the indicated workload parameters, where

AVGSKILL	=	average skill (1-low to 5-high) of support personnel
PTCORR	=	percentage of change requests which are corrections
PCLOW	=	percentage of change requests which are low complexity
PCHIGH	=	percentage of change requests which are high complexity
PPNORM	=	percentage of change requests which are normal priority
TYPE	=	1 if system type is same as TYPE; 0 otherwise.

The estimated risk that the available productivity will not be sufficient for the expected workload is the area under the part of a normal distribution for person months per change larger than the available person months per change.

COMPUTATION OF ESTIMATED RISK
(BLOCK 2 RELEASE)

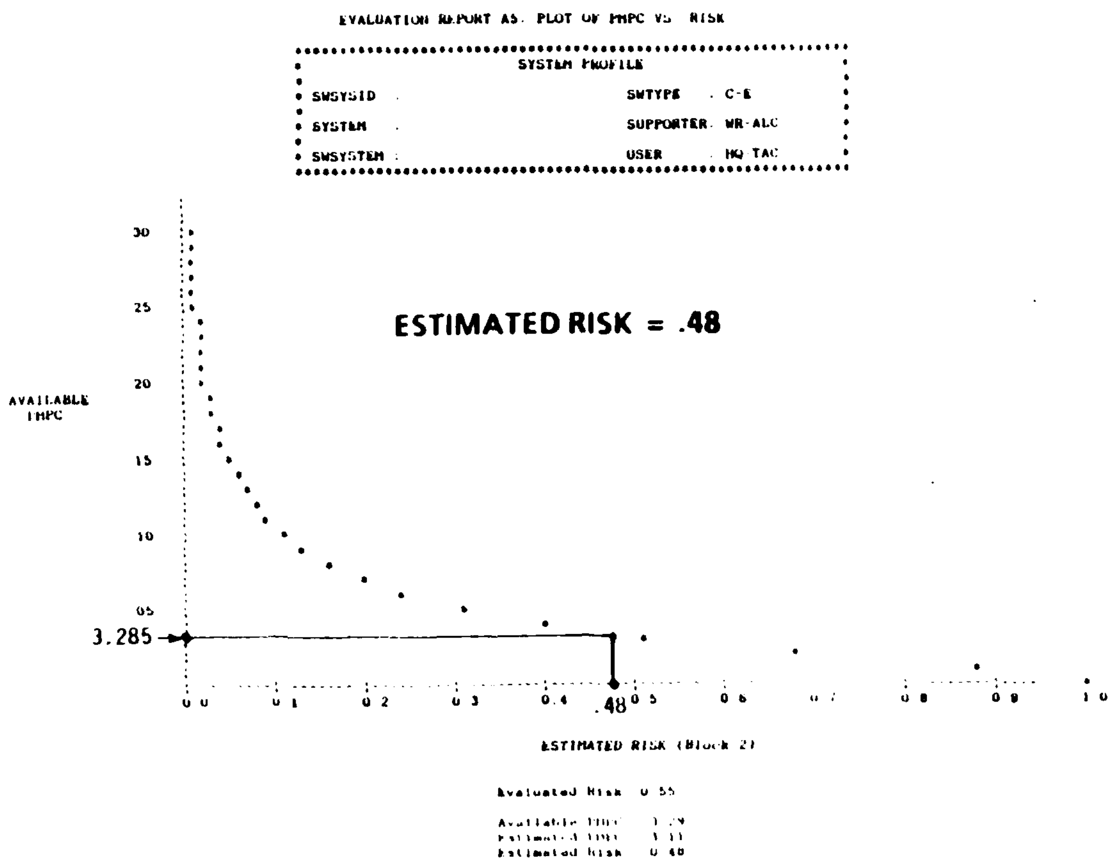
AVAILABLE PRODUCTIVITY

(15* .19 + 9* .9) (0.667)(9)/20 = 3.285

ESTIMATED PRODUCTIVITY

AVGSKILL	3.00
PTCORR	0.75
PCLOW	0.75
PCHIGH	0.05
PCNORM	0.75
TYPE	C-E
L\	1.13
PMPC\	3.11

ESTIMATED RISK = .48



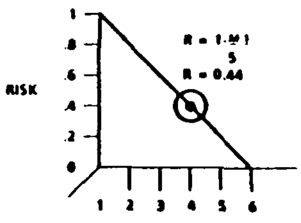
The computations for estimated productivity, available productivity, and estimated risk are outlined on this slide for the block 2 release. The estimated productivity is computed using the regression equation previously mentioned and determines the normal distribution curve which applies to this case. The available productivity is computed from the workload data obtained from the user/supporter baseline estimate. Given the normal distribution function and available productivity, the estimated risk is the integral of the area under the distribution curve to the right of the available productivity. Alternatively, the slide shows an output from the RAMSS support software which provides a cumulative distribution function. The vertical axis plots available person months per change, and the horizontal axis is estimated risk. This plot allows the analyst to examine tradeoffs for how much improving available productivity may reduce risk, or how much available productivity must be improved to meet a specific risk.

EVALUATED SUPPORTABLILITY RISK

SUPPORTABILITY METRICS

PL	OS	DM	IM	TS	PI	ID	CO	SA	RA	MO	DS	CO	SI	EX	IN	MO	DS	CO	SI	EX	IN	MG	TE	SU	CO	HO	BE	LA	OP	CM	OT	GE	SU
3.33	3.33	4.00	3.50	3.67	3.00	3.33	3.50	3.00	2.83	4.70	3.50	3.80	3.90	3.40	4.50	5.20	4.20	3.50	5.00	4.60	3.40	3.75	3.38	3.17	3.83	4.02	4.07	0.00	4.11	2.92	3.50	4.50	4.67
PROJECT MANAGEMENT						CONFIGURATION MANAGEMENT				DOCUMENTATION						SOURCE CODE						PERSONNEL				SUPPORT SYSTEMS						FAC	
3.47						3.17				3.97						4.32						3.53				3.72						4.58	
SOFTWARE LIFE CYCLE PROCESS										SOFTWARE PRODUCT												SOFTWARE SUPPORT RESOURCES											
3.32										4.15												3.94											
SOFTWARE SUPPORTABILITY																																	
3.80																																	

SUPPORTABILITY RISK COMPUTATION



LINEAR MODEL
(INITIAL APPROXIMATION)

L^ = 4.90401 - 0.29131 (PRODUCT) - 0.15600 (PERSONNEL)
- 0.25120 (SYSTEMS) + 0.04294 (FACILITIES)
- 0.66174 (PROCESS)

$$R\hat{} = \left[\frac{1}{1 + e^{-L\hat{}}} - .01 \right] / (0.98)$$

R^ = 0.55

CURRENT MODEL

SUPPORTABILITY RISK DRIVERS

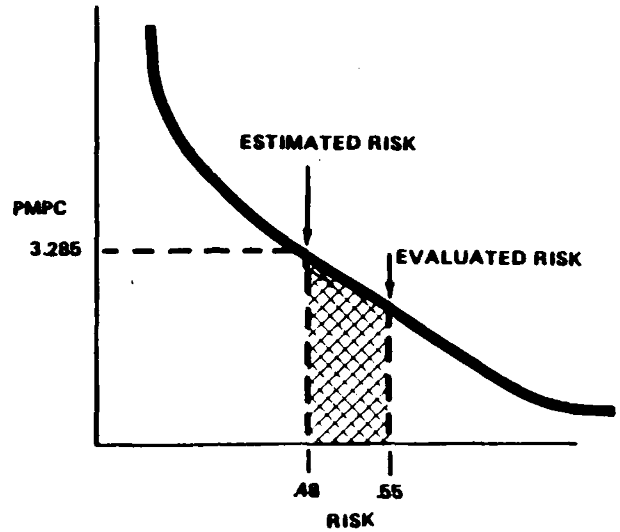
SW LIFE CYCLE PROCESS
CONFIGURATION MANAGEMENT = 3.17
PROJECT MANAGEMENT = 3.47

SW RESOURCES
PERSONNEL = 3.53

The next step in the risk assessment methodology is the application of the supportability evaluation methods previously discussed. As shown, the scores on a 1 to 6 scale from each sublevel of evaluation are combined at the next highest level. The overall software supportability score of 3.8 was converted to risk by means of the linear model shown as an initial approximation. A more realistic linear regression model is being currently used. In this model, the five covariate factor values as indicated are substituted into the equation. The evaluated risk in this example is .55. The supportability risk drivers can be determined by substituting perfect scores for each of the factors separately and computing the resulting amount of risk reduction possible.

ESTIMATED AND EVALUATED RISK INTEGRATION

- DISTRIBUTION FUNCTIONS



BLOCK 2

- RISK COMPARISON

	BLOCK 1	BLOCK 2	BLOCK 3
ESTIMATED RISK	0.14	0.48	0.54
EVALUATED RISK	0.55	0.55	0.55
ADDITIONAL RISK	0.41	0.07	0.01

Given that the "evaluated" risk is .55, this number is plotted on the cumulative distribution. It is now possible to see the difference between the "estimated" risk and the "evaluated" risk. What this says is that certain characteristics of the current software are making the risk higher than what the user and supporter estimated the risk to be. This difference is shown in the risk comparison table for all block releases. The question now becomes, what do you do about it?

TRADEOFF ANALYSIS

TRADEOFF CATEGORY	EXAMPLE TRADEOFF	RISK IMPACT (BLOCK 2)	EFFECTIVE?
ESTIMATED RISK			
AVAILABLE PERSONS	9 PERSONS (90% DED) ⇒ 12 PERSONS (90% DED)	ESTIMATED RISK: 0.48 ⇒ 0.39	YES
BLOCK RELEASE SCHEDULE	9 MO/3 MO OVERLAP ⇒ 12 MO/NO OVERLAP	ESTIMATED RISK: 0.48 ⇒ 0.22	YES
CHANGE REQUEST PROFILE	20: (15,5,0); (1,4,15); (1,4,15) 20: (18,2,0); (1,2,17); (0,2,18)	ESTIMATED RISK: 0.48 ⇒ 0.39	YES
EVALUATED RISK			
SUPPORTABILITY METRIC ACCURACY	VARIANCE OF ± 10% IN FIVE FACTORS PRODUCT: 4.15 ⇒ (4.57, 3.74) PERSONNEL: 3.53 ⇒ (3.88, 3.18) SYSTEM: 3.72 ⇒ (4.09, 3.35) FACILITIES: 4.58 ⇒ (5.04, 4.12) LIFE CYCLE: 3.32 ⇒ (3.65, 2.99)	EVALUATED RISK: 0.55 ⇒ (0.43, 0.67)	MAYBE
SUPPORTABILITY RISK DRIVERS	CASE 1: SCM 3.17 ⇒ 4.58 CASE 2: SLCP 3.32 ⇒ 4.58 CASE 3: SUPPORT PERSONNEL 3.53 ⇒ 4.58 CASE 4: CASES 1, 2, &3	CASE 1: 0.55 ⇒ 0.43 CASE 2: 0.55 ⇒ 0.35 CASE 3: 0.55 ⇒ 0.51 CASE 4: 0.55 ⇒ 0.31	YES YES NO YES
BASELINE ESTIMATE CHANGE	AS IN ESTIMATED RISK ABOVE CASE 1: AVAILABLE PERSONS CASE 2: BLOCK RELEASE SCHEDULE CASE 3: CHANGE REQUEST PROFILE	CASE 1: 0.55 ⇒ 0.55 CASE 2: 0.55 ⇒ 0.55 CASE 3: 0.55 ⇒ 0.54	NO NO NO

This slide shows various methods which may be applied to reduce the risk. Some of the effects listed for this particular example are significant, and are representative of the fact that it is possible to quantify areas which have the biggest payoff. Various combinations of these trade-offs may be applied to reduce the risk to acceptable levels.

WHEN CAN RAMSS BE APPLIED?

<u>Anytime</u> During the Software System Life Cycle (From Concept Development to Production)

Note:

The collection of Data to Support the Software Life Cycle Process (SLCP) Evaluation is Continuous

Recommended Points for Assessing Risk:

- Milestone 0
- Milestone I (Functional Baseline)
- Milestone II (Allocated Baseline)
- Milestone III (Production Baseline)

There are several significant benefits to the RAMSS, but a big advantage is the capability to apply risk assessment at any point during the software system life cycle. Obviously in the early stages of system concept definition and development there may be little data available about software support requirements. But the RAMSS user can still input whatever data is available, or whatever best estimates might be, to assess the risk of software supportability. As the system matures, more detailed information will be available for both estimating risk and computing evaluated risk, hence making the overall risk assessment more accurate. However, there are potential high payoffs from being able to identify risk drivers as early as possible. Recommended points for assessing risk are shown.

WHAT GOVERNMENT RESOURCES ARE REQUIRED TO SUPPORT THE RAMSS?

EQUIPMENT:

AFOTEC – IBM PC/AT (or Compatible) with 640K RAM and Bernoulli Box

Floppy Disk Drive	dBASE III (Data Base Management System)
Monochrome Display	BMDP/PC (Statistical Software Package)
Dot Matrix Printer	MS-DOS (Operating System)

(Cost ~ $8,000)

OTHER – None

PERSONNEL:

AFOTEC – RAMSS Support and Use: ~ 1/2 Person Per Year

Software Evaluations: ~ 1/4 Person for Length of AFOTEC Weapon System Involvement

OTHER – Supporting Command: ~ 2 person days/software block release (.1 person days if information available from an automated configuration management system)

~ 2 person days to determine user/supporter baseline estimate

Using Command: ~ 2 person days to determine user/supporter baseline estimate

(Software Evaluations, Supporting Command and Using Command: Per Evaluated Weapon System)

There was a considerable attempt during the development of the RAMSS to minimize its own support requirements. Equipment and personnel support estimates are shown in this slide. The most important personnel requirement is the dedication of an AFOTEC individual as a focal point for the RAMSS support and use. Note that support requirements from the Supporting Command can be significantly reduced given that an automated configuration management tool is available to provide the historical data which will keep the RAMSS a current and valuable tool. A few automated configuration management tools exist in the Supporting Commands visited during the RAMSS development. However, the vast majority of the configuration management processes were either manually accomplished or were not collecting data useful to the RAMSS.

WHAT ARE THE FUTURE COMMITMENTS NECESSARY FOR RAMSS TO MATURE?

- Continue to Collect Software Maintenance Activity Data (Release Schedule, Support Personnel, Software Change Requests) for All Systems Block Releases
- When Possible, Collect Actual Software Maintenance Activity Data (This Data was Not Available During the RAMSS Development)
- Periodically Update RAMSS Regression Equations Used to Compute Software Supportability Risk
- Assignment of an AFOTEC RAMSS System Manager
- Continued Assessment and Update of the AFOTEC Software Evaluation Tools
  a. Software Life Cycle Process
  b. Software Product Maintainability
  c. Software Support Resources

The RAMSS has been developed, but is not mature. The method will only be as good as the historical data from which it can estimate risk. Hence, to keep the RAMSS current and useful, it is important that AFOTEC continue to collect data on software maintenance activity for block releases from current and new software systems. In addition, the methodology would be improved by a capability to collect actual resources used per software change request rather than available resources per software change request. As a result of collecting new data, the regression equations upon which the risk is based may require periodic update. *Finally, the tools used to evaluate the* software supportability should be assessed and updated as appropriate.

SUMMARY

- RISK ASSESSMENT CAN BE APPLIED TO SOFTWARE SUPPORTABILITY
- THE METHODOLOGY BEING GENERATED MAY HAVE DIRECT APPLICATION TO OTHER T&E PROBLEMS

In summary, we have reviewed the purpose, background, and methodology of the RAMSS, and demonstrated its application to the software supportability problem. It is entirely possible that the approach taken by the RAMSS may have direct application to other test and evaluation problems. This application will require further study.

Appendix B
RAMSS: Evaluator's Guide

APPENDIX B

RISK ASSESSMENT METHODOLOGY FOR SOFTWARE SUPPORTABILITY (RAMSS): EVALUATOR'S GUIDE

a. The purpose of this appendix is to provide the Software Test Manager (STM) and Deputy for Software Evaluation (DSE) with the information needed to accomplish the Air Force Operational Test and Evaluation Center's (AFOTEC's) software supportability risk assessment. The accumulation of procedures, analysis, and methodology is denoted as the Risk Assessment Methodology for Software Supportability (RAMSS).

b. This appendix is an evolutionary document that should be updated periodically. The form of the risk assessment is dependent upon the current AFOTEC software supportability evaluations, the historical data base of software supportability evaluations, and the historical data base of software maintenance release data.

c. This appendix is intended to be a volume in a series of Software Operational Test and Evaluation Guidelines prepared by the Software Evaluation Division of the Logistics Directorate. It is intended for use in the operational test and evaluation of software. Comments should be directed to the Office of Primary Responsibility (OPR). The series of guidelines are:

(1) AFOTEC Pamphlet 800-2, Volume 1--Management of Software Operational Test and Evaluation

(2) AFOTEC Pamphlet 800-2, Volume 2--Reserved

(3) AFOTEC Pamphlet 800-2, Volume 3--Software Maintainability - Evaluator's Guide

(4) AFOTEC Pamphlet 800-2, Volume 4--Software Operator-Machine Interface - Evaluator's Guide

(5) AFOTEC Pamphlet 800-2, Volume 5--Software Support Facility Evaluation - User's Guide

(6) AFOTEC Pamphlet 800-2, Volume 6--Reserved.

d. Additional documents required to understand the RAMSS and its application include:

(7) RAMSS User's Handbook, BDM/ABQ-85-1270-TR
(8) RAMSS Software Life Cycle Evaluator's Guide, BDM/ABQ-86-0090-TR, Appendix A

This Guide is organized as follows:

APPENDIX B

ATTACHMENTS

FIGURES

FIGURES (Concluded)

B.1 GENERAL.

a. Software supportability is a measure of the adequacy of personnel, resources, and procedures to facilitate the support activities of modifying and installing software, establishing an operational software baseline, and meeting user requirements. Software supportability is a function of the quality of the software products, the capabilities of the software support resources, and the life cycle management processes which control the procurement, development, operation, and support of the software.

b. The software supportability risk is the likelihood that the Air Force supporting command will not be able to accomplish the necessary support of the software with planned or actual support resources.

c. The focus of this guide is upon the process which the responsible evaluator should apply in order to derive the software supportability risk.

B.2 OVERVIEW OF METHODOLOGY: RESPONSIBILITY, USE, RESULTS.

a. The RAMSS evaluator will usually be the STM and/or the DSE. The STM/DSE should read paragraphs B.1 through B.9 in their entirety and understand the RAMSS concept and procedures before beginning any risk assessment. These pages provide the evaluator with:

(1) A background of the RAMSS development

(2) A basic understanding of the RAMSS procedures

(3) Detailed instruction for use of the RAMSS automated tool support capabilities for analysis and reporting requirements.

b. The RAMSS uses the results from the AFOTEC software supportability evaluations of the software life cycle process, software product maintainability, and software support resources, along with historical software evaluation and maintenance release data, to determine the software supportability risk. In addition, analysis reports enable the evaluator to determine which supportability factors are rated low relative to the historical evaluation data, and which supportability factors have the most impact on the software supportability risk. Guidelines are presented to enable the evaluator to classify the software supportability risk as high, medium, or low. The high-level flow of the RAMSS is illustrated in figure B-1. The software supportability evaluation hierarchy is shown in figure B-2 to the level required by RAMSS.

c. All required input data and output analysis reports for RAMSS are managed by the RAMSS automated support system described in the RAMSS User's Handbook. A functional flow of the automated support system is shown in figure B-3. The RAMSS automated support system is menu-driven and uses IBM-PC/AT or compatible hardware, a dBase III data base management system, and BMDP statistical software. The basic functions of the automated support system for RAMSS include:

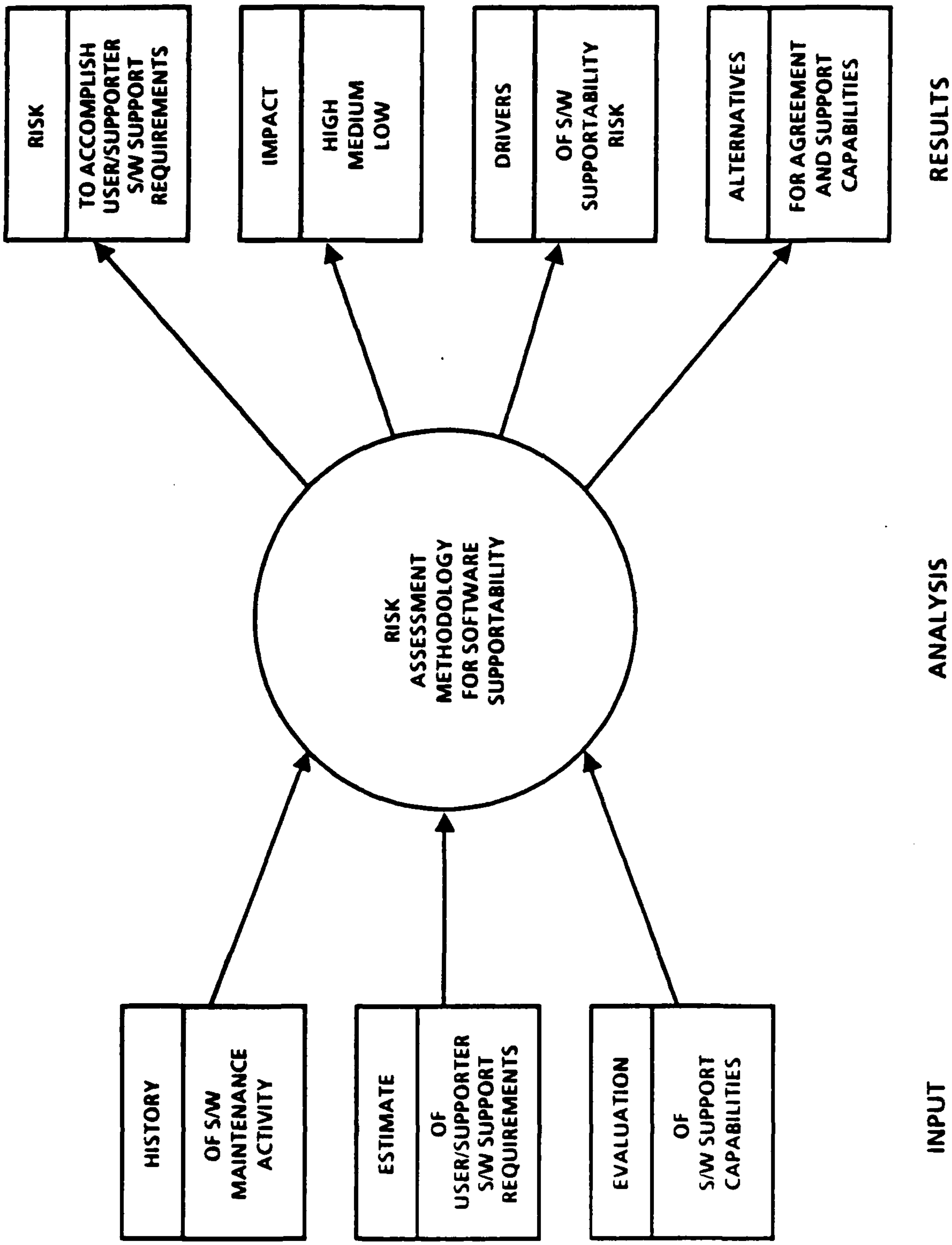


Figure B-1. RAMSS High Level Data Flow

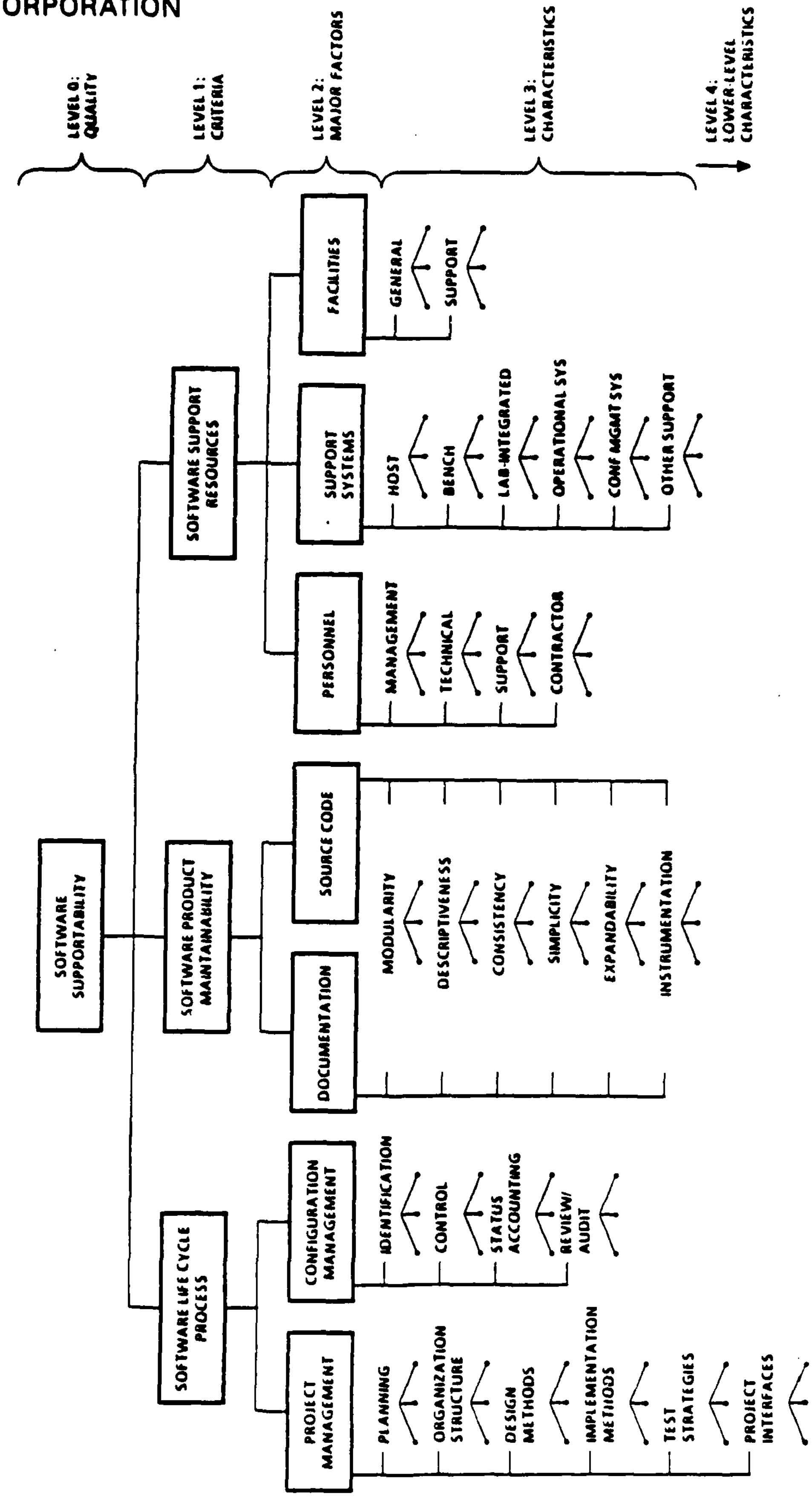


Figure B-2. Elements of Software Supportability Evaluations

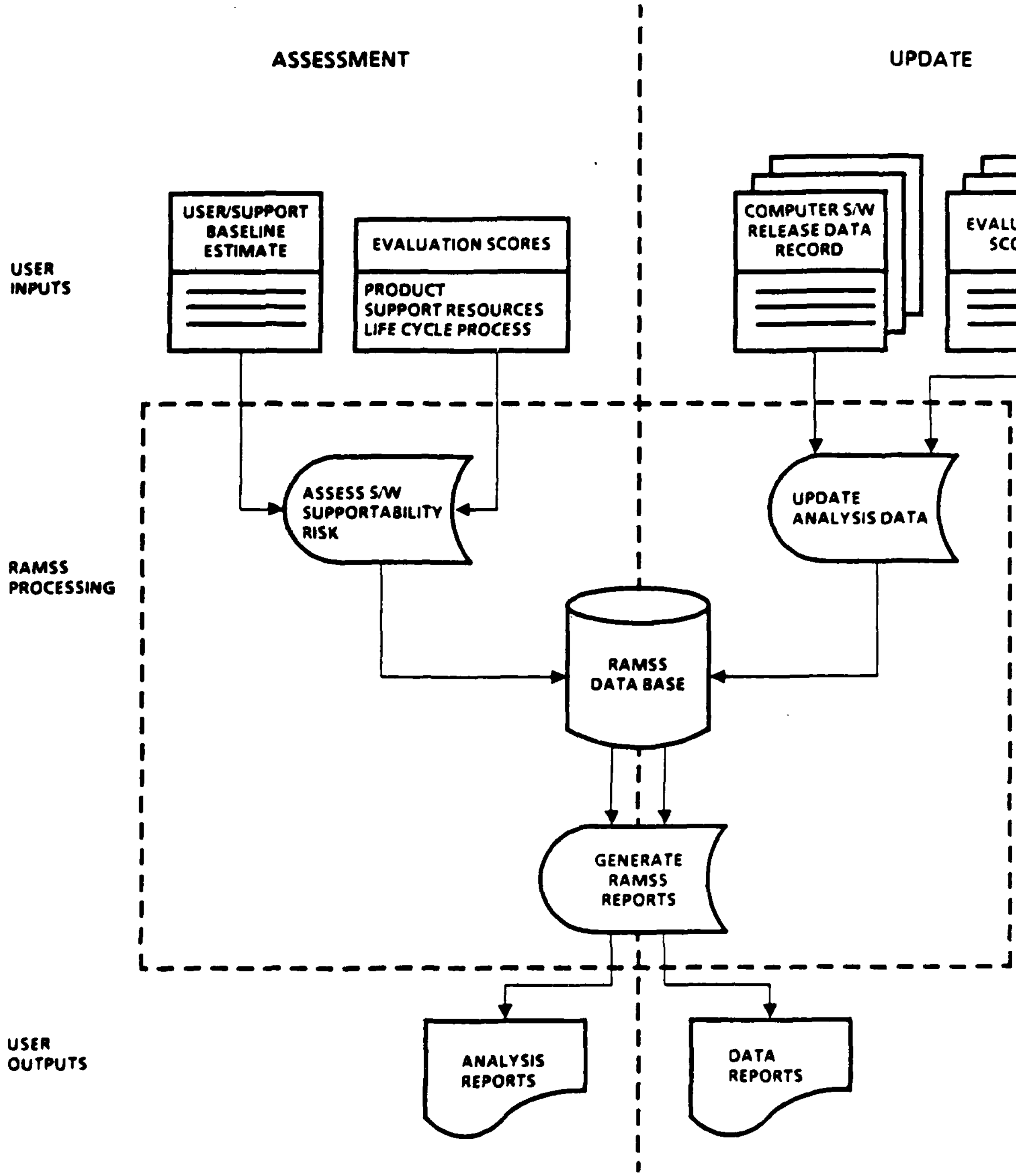


Figure B-3. Overview of Major Processes of RAMSS: Risk Assessment and Analysis Data Update

(1) Entry/update/report of evolving user/supporter baseline workload estimate of software support resources, block release change profiles, and estimated risk

(2) Entry/update/report of software supportability evaluation results and evaluated risk

(3) Output of various dBase III analysis reports

a) Report of software supportability evaluation showing percentile of evaluation ratings relative to all other systems in the historical data base

b) Report of software supportability evaluation risk reduction drivers

c) Report plot of workload in person months per change versus risk

d) Summary report of important risk assessment results

(4) Entry/update/report of historical evaluation and maintenance release data

a) dBase III data base reports

b) BMDP statistical analysis reports.

d. The RAMSS automated support system interfaces are through console menu selection and data entry, and output reports generated on the printer. The system is very simple. The system does not provide a wide variety of automated "what if" analysis or custom reports. Its focus is upon providing a basic capability to enter evaluation data, receive an assessment of the associated software's supportability risk through printed reports, and update the necessary historical data bases.

e. Detailed requirements for use of the RAMSS automated support system are described in the RAMSS User's Handbook.

B.3 PROCEDURE FOR APPLYING RAMSS TO AFOTEC PROGRAMS.

a. Application of RAMSS to AFOTEC programs is appropriate whenever the system contains significant or critical software systems for which Air Force software support during post-deployment support of the system is required.

b. Risk assessment of software supportability is a life cycle process. There are key points (such as milestones 0, 1, 2, 3, critical design review, IOC, PMRT) throughout a software system's life cycle where application of a RAMSS (or some part of it) would be beneficial. Benefits which might occur include: early planning and trade-off studies for software support resource requirements; early view of potential software support management problems; early visibility of user requirements for expected software support actions; capability to trace software supportability risk profile (i.e., measures of risk) throughout the life cycle; early view of expected software supportability risk drivers; and the actual assessment of the risk to user and supporter which must be accepted before support of the software can be assumed.

c. The general RAMSS procedure is illustrated in figure B-4. The application of RAMSS throughout the software life cycle process as integrated with AFOTEC OT&E phases and functions is shown in figure B-5. This chart illustrates the areas of emphasis for AFOTEC involvement using the RAMSS from early concept exploration through post-deployment support. These areas of emphasis reflect the tailoring of the software supportability evaluations from which results will be input to the RAMSS.

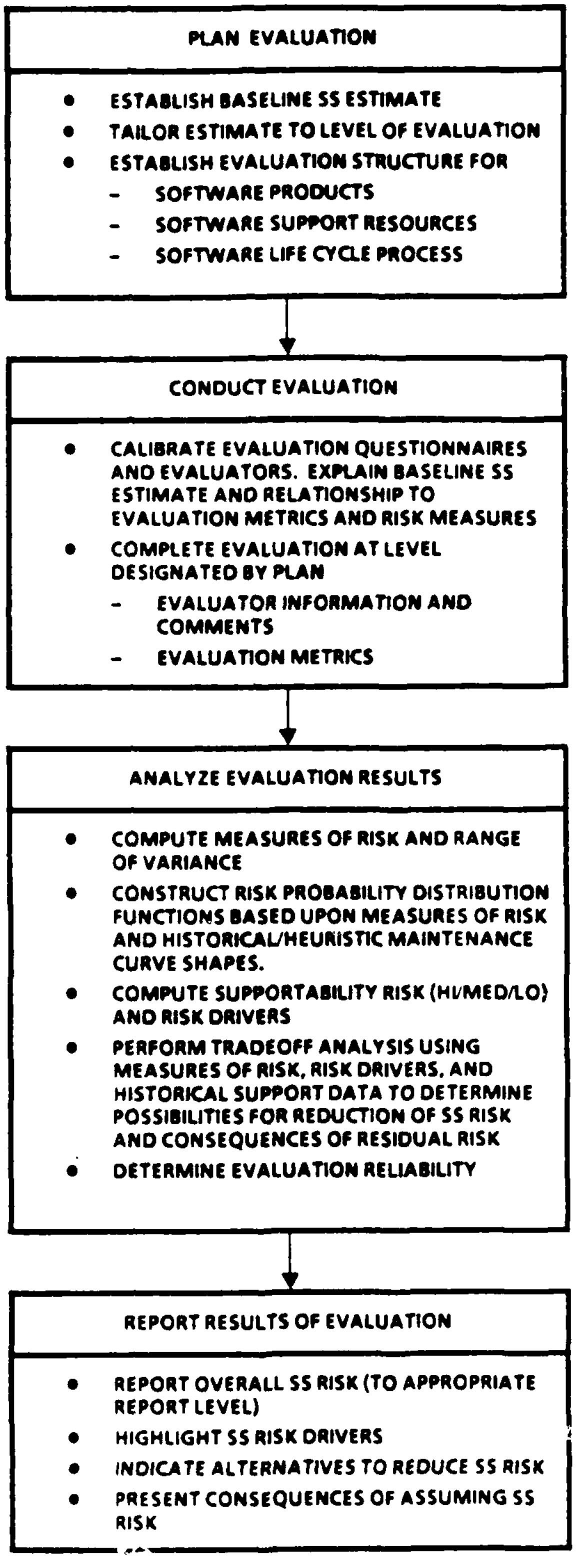
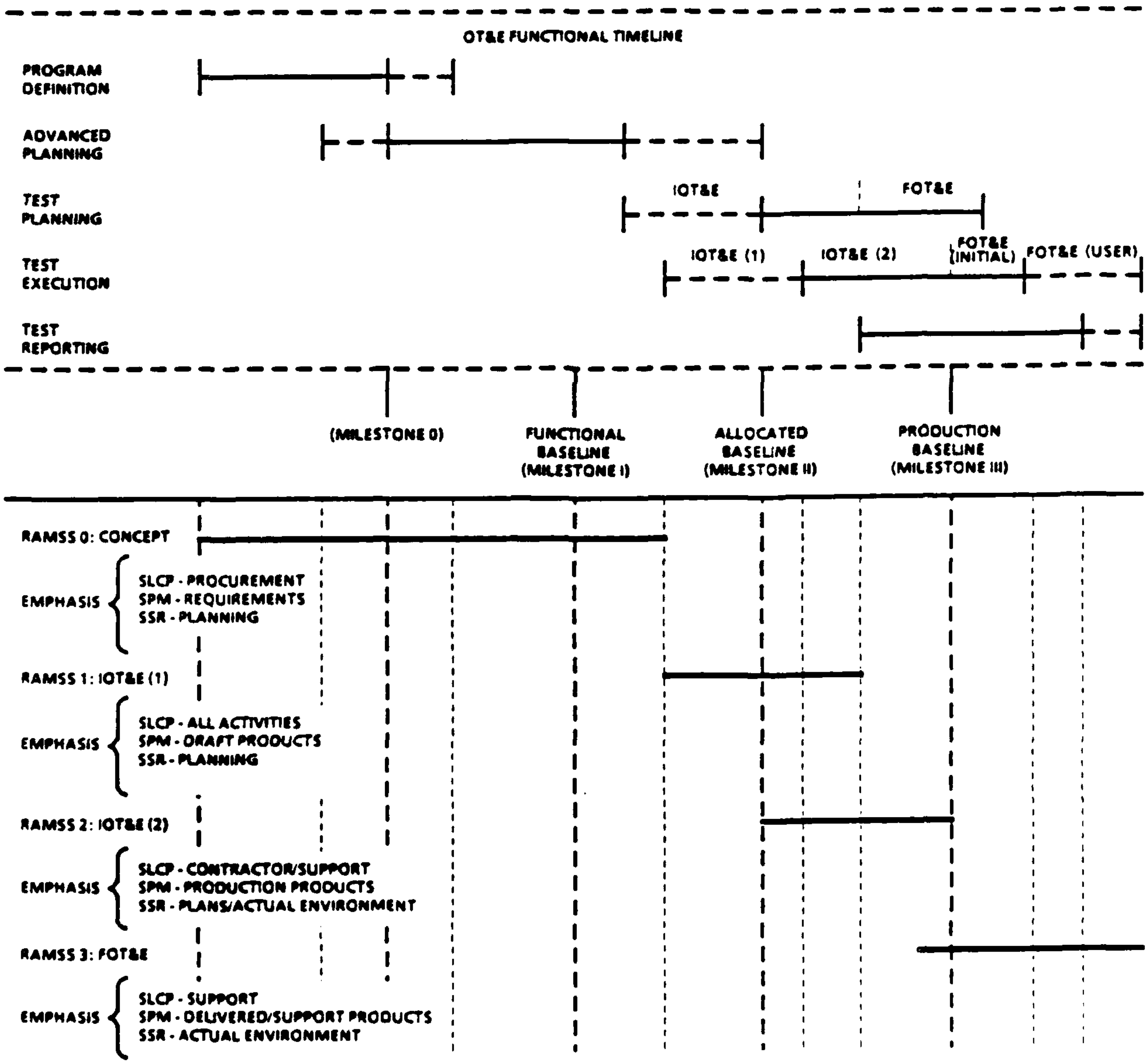


Figure B-4. Integration of RAMSS and the Software Supportability Evaluation Process

800 SERIES (WEAPON)	MISSION ANALYSIS	CONCEPT EXPLORATION	DEMONSTRATION AND VALIDATION	FULL SCALE DEVELOPMENT	PRODUCTION AND DEPLOYMENT
300 SERIES (AUTOMATIC DATA PROCESSING)	PLANNING	CONCEPT	DEFINITION	DEVELOPMENT ¦ TEST	OPERATION AND MAINTENANCE
MAJOR OT&E ACTIVITY	· TRACK PROJECTED PROGRAMS · REVIEW CRITICAL ISSUES · PREPARE INPUTS TO PMD	· ESTABLISH REQUIREMENTS FOR OT&E OF SYSTEM · INITIATE RISK ANALYSIS · PREPARE OT&E INPUTS TO TEMP	· EVALUATE OT&E TRADEOFFS/ OPTIONS · REFINE OT&E OBJECTIVES/ SUBOBJECTIVES · MONITOR/CONDUCT T&E ON PROTOTYPE SYSTEM · PREPARE INPUTS TO CRISP AND UPDATED TEMP	· MONITOR PDR/CDR · UPDATE OT&E PLANS (TEMP, CRISP, O/S CMP, TPO, IOT&E) · CONDUCT IOT&E · UPDATE INPUTS TO OPERATIONS AND SUPPORT CONCEPTS · REPORT IOT&E RESULTS	· MONITOR ESTABLISHMENT OF OPERATIONAL AND SUPPORT ENVIRONMENTS · UPDATE FOT&E PLANS · CONDUCT FOT&E · REPORT FOT&E RESULTS



SLCP - SOFTWARE LIFE CYCLE PROCESS
SPM - SOFTWARE PRODUCT MAINTAINABILITY
SSR - SOFTWARE SUPPORT RESOURCES

Figure B-5. Application of RAMSS for AFOTEC OT&E

d. Specific evaluator guidance for evaluation of a user/supporter baseline estimate, integration of software supportability evaluation results, generation of analysis reports, and reporting the risk assessment results is contained in the next sections.

B.4 DERIVING A USER/SUPPORTER BASELINE ESTIMATE.

a. The user/supporter baseline estimate is simply an estimation of the support resources and software change activity expected for a given software system for one or more block releases during post-deployment software support. This estimate is derived by reviewing historical software maintenance data, available acquisition planning information in documents such as the CRISP or O/S CMP, the current software system status, and the perspective of the using and supporting command personnel.

b. The process of deriving a baseline estimate may iterate until a reasonable consensus or compromise is reached among the using and supporting command personnel, and AFOTEC STM/DSE personnel. The basic four steps, which may be repeated, are shown in figure B-6.

STEP 1:	DERIVE DRAFT OF ESTIMATE USING THE RAMSS AUTOMATED SUPPORT SYSTEM
STEP 2:	OBTAIN REVIEW COMMENTS ON THE DRAFT FROM USING COMMAND PERSONNEL
STEP 3:	OBTAIN REVIEW COMMENTS ON THE DRAFT FROM SUPPORTING COMMAND PERSONNEL
STEP 4:	WORK OUT COMPROMISE BETWEEN USING AND SUPPORTING COMMAND ON DRAFT AND UPDATE NEXT DRAFT ON RAMSS AUTOMATED SUPPORT SYSTEM

Figure B-6. User/Supporter Baseline Estimate Evolution Steps

c. The inputs for the baseline estimate are:

(1) System description names

(2) Support resources in form of release schedule and support personnel (full-time equivalents and skill level)

(3) Block release change profile (number, type, complexity, priority) for up to three blocks (manual input and input from maintenance release data or other baseline estimates is possible).

d. The outputs of the baseline estimate computations for each block release are:

(1) Available person months per change

(2) Estimated (optimum) person months per change

(3) Estimated software supportability risk based upon the baseline estimate workload parameters.

Threshold and goal values of 0.50 and 0.20 are reasonable boundaries for defining high, medium, and low risk.

e. An example report of a user/supporter baseline estimate is shown in figure B-7. Details for use of the RAMSS automated support system can be found in the RAMSS User's Handbook.

B.5 INTEGRATING SOFTWARE SUPPORTABILITY EVALUATION RESULTS.

a. The steps to integrating the software supportability evaluation scores in order to derive the evaluated software supportability risk assessment results are:

EVALUATION REPORT A1: USER /SUPPORTER BASELINE CONCEPT

```
*****************************************************************
*                        SYSTEM PROFILE                          *
*                                                                *
* SWSYSID  :                           SWTYPE    : C-E           *
*                                                                *
* SYSTEM   :                           SUPPORTER: WR-ALC         *
*                                                                *
* SWSYSTEM :                           USER      : HQ-TAC        *
*****************************************************************
```

SUPPORT CONCEPT

Release Schedule: 9.00 Month block release cycle with 3.00 Month overlap

Support Staff : 15 Persons, 19% Dedicated, Avg.skill level 3.00

9 Persons, 90% Dedicated, Avg.skill level 3.00

BASELINE SUPPORT PROFILE

BLOCK	TOTAL # CHANGES	TYPE(C,H,V)	COMPLEXITY(H,M,L)	PRIORITY(E,U,N)
1	15	(15, 0, 0)	(0, 0, 15)	(0, 0, 15)
2	20	(15, 5, 0)	(1, 4, 15)	(1, 4, 15)
3	20	(13, 7, 0)	(1, 6, 13)	(1, 6, 13)

ESTIMATED SOFTWARE SUPPORTABILITY RISK

BLOCK	AVAILABLE PERSON MONTHS PER CHANGE	ESTIMATED PERSON MONTHS PER CHANGE	ESTIMATED RISK
1	5.47	1.98	0.14
2	3.29	3.11	0.48
3	3.29	3.57	0.54

Figure B-7. Example Report A1: User/Supporter Baseline Concept

(1) Step 1: Obtain the following 34 evaluation characteristics scores from the software supportability evaluations:

a) Documentation: Modularity, Descriptiveness, Consistency, Simplicity, Expandability, Instrumentation

b) Source Listings: Modularity, Descriptiveness, Consistency, Simplicity, Expandability, Instrumentation

c) Personnel: Management, Technical, Support, Contractor

d) Support Systems: Host, Software Bench, Laboratory Integrated Test, Operational Integrated Test, Configuration Management System, Other

e) Facilities: General Office Space, Support Systems Environment

f) Project Management: Planning, Organization Structure, Design Methods, Code/Implementation Methods, Test Strategies, Project Interfaces

g) Configuration Management: Identification, Control, Status Accounting, Audit/Review.

(2) Step 2: In addition to the 34 evaluation scores of step 1, entry is required of an important overall assessment score which is called the software supportability confidence. On the basis of all available evaluation data, software system review information, working group data, and so forth, the software test manager/deputy for software evaluation assesses the confidence that the subject software system can be supported at the

level of activity indicated by the user/supporter baseline estimate. This is a value between 0 (low) and 1 (high). It is only used as part of the future risk regression equation update process. See the RAMSS User's Handbook for further information on the update process. The confidence value does not affect results of the current evaluation.

(3) Step 3: Enter the evaluation scores into the RAMSS data base. The user enters the 34 evaluation scores plus the confidence assessment score into the RAMSS evaluation data base. If desired, the low-level software life cycle process evaluation scores (see RAMSS Software Life Cycle Evaluator's Guide and RAMSS User's Handbook) can be entered instead of the ten level 3 characteristic scores. The screen input format is illustrated in figure B-8.

```
                     RAMSS 03/14/86 SCREEN 1.3.1.1
SYSTEM:              SWSYSTEM:                    SWTYPE: C-E    SWSYSID:
     RAMSS SOFTWARE SUPPORTIBILITY EVALUATION SCORES

*LIFE CYCLE PROCESS 3.32       *PRODUCT 4.15          *SUPPORT RESOURCES 3.94
  PROJECT MANAGEMENT 3.47        DOCUMENTATION 3.97      PERSONNEL 3.53
    Planning                   3.33 Modularity       4.70   Manager        3.75
    Organizational Structure   3.33 Descriptiveness  3.50   Technical      3.38
    Design Methods             4.00 Consistency      3.80   Support        3.17
    Implementation Methods     3.50 Simplicity       3.90   Contractor     3.83
    Test Strategies            3.67 Expandability    3.40
    Project Interface          3.00 Instrumentation  4.50 SUPPORT SYSTEMS 3.72
                                                            Host           4.02
  CONFIGURATION MANAGEMENT 3.17 SOURCE LISTINGS  4.32       Bench          4.07
    Identification        3.33    Modularity         5.20   Lit            0.00
    Configuration Control 3.50    Descriptiveness    4.20   Oit            4.11
    Status Accounting     3.00    Consistency        3.50   CMS            2.92
    Audit                 2.83    Simplicity         5.00   Other          3.50
                                  Expandability      4.60 FACILITIES 4.58
** Computed Overall Score 3.80    Instrumentation    3.40   General        4.50
** Evaluated Risk 0.55                                      Support Sys.   4.67
       S/W Supportability Confidence Assessment 0.70
ENTER OPTION (E-EDIT; S-SAVE; W-WHAT IF; R-RETURN; Q-QUIT)
```

Figure B-8. RAMSS Screen Entry of Software Supportability Evaluation Results

(4) Step 4: Compute necessary hierarchical evaluation scores and associated risk values. This step is conducted partially when the evaluation data is entered, and partially when RAMSS printed reports are generated. This step does not require any direct evaluator participation other than generating the reports through menu selection.

B.6 OBTAINING RISK ASSESSMENT RESULTS.

The software supportability risk assessment results are contained in six dBase III reports and seven BMDP reports which can be generated through the RAMSS automated support system. Each of these reports is briefly described in the following paragraphs. Examples of the dBase III analysis reports are illustrated. The BMDP example reports and further interpretation of all example reports are in the RAMSS User's Handbook.

B.6.1 dBase III Risk Assessment Analysis Reports. There are six possible dBase III reports which contain risk assessment results. In addition, there are five raw data reports which are essentially formatted reports of all the data in the historical evaluation and maintenance release data bases, and various analysis parameters derived from the data bases.

B.6.1.1 Report A1: User/Supporter Baseline Estimate. This report (see example in figure B-7) contains the baseline estimate inputs as well as the computed available person months per change, estimated (optimal) person months per change, and the estimated software supportability risk for each of up to three block releases. This report is used as an input to the software supportability evaluations, and to perform trade-off analysis for support resources (personnel, skill level, and release cycle) and the baseline change profiles (number, type, complexity, priority of block release changes). The estimated person months per change is computed from a linear regression model using support resources and baseline change

EVALUATION REPORT A2: TABLE OF EVALUATION SCORES

```
*****************************************************************
*                         SYSTEM PROFILE                        *
*                                                               *
* SWSYSID  :                          SWTYPE    : C-E           *
*                                                               *
* SYSTEM   :                          SUPPORTER: WR-ALC         *
*                                                               *
* SWSYSTEM :                          USER      : HQ-TAC        *
*****************************************************************
```

I. LIFE CYCLE PROCESS 3.32

- A. PROJECT MANAGEMENT 3.47
  - 1. Planning 3.33
  - 2. Organizational Structure 3.33
  - 3. Design Methods 4.00
  - 4. Implementation Methods 3.50
  - 5. Test Strategies 3.67
  - 6. Project Interface 3.00
- B. CONFIGURATION MANAGEMENT 3.17
  - 1. Identification 3.33
  - 2. Configuration Control 3.50
  - 3. Status Accounting 3.00
  - 4. Audit 2.83

II. PRODUCT 4.15

- A. DOCUMENTATION 3.97
  - 1. Modularity 4.70
  - 2. Descriptiveness 3.50
  - 3. Consistency 3.80
  - 4. Simplicity 3.90
  - 5. Expandability 3.40
  - 6. Instrumentation 4.50
- B. SOURCE LISTINGS 4.32
  - 1. Modularity 5.20
  - 2. Descriptiveness 4.20
  - 3. Consistency 3.50
  - 4. Simplicity 5.00
  - 5. Expandability 4.60
  - 6. Instrumentation 3.40

III. SUPPORT RESOURCES 3.94

- A. PERSONNEL 3.53
  - 1. Manager 3.75
  - 2. Technical 3.38
  - 3. Support 3.17
  - 4. Contractor 3.83
- B. SUPPORT SYSTEMS 3.72
  - 1. Host 4.02
  - 2. Bench 4.07
  - 3. Laboratory 0.00
  - 4. Operational 4.11
  - 5. CMS 2.92
  - 6. Other 3.50
- C. FACILITIES 4.58
  - 1. General 4.50
  - 2. Support Systems 4.67

Computed Overall Score (Average of I, II, and III): 3.80

Software Supportability Confidence Assessment : 0.70

Software Supportability Evaluated Risk : 0.55

Figure B-9. Example Report A2: Table of Evaluation Scores profile parameters (see section B.7). The estimated values can be used for early computer resources planning.

B.6.1.2 Report A2: Table of Evaluation Scores. This report (see example in figure B-9) contains all input and accumulated software supportability evaluation scores for levels 3, 2, 1, and 0 of the evaluation hierarchy. In addition, the evaluated software supportability risk is output. This evaluated risk is computed from a linear regression model using these five factors: software life cycle process, software product maintainability, support personnel, support systems, and support facilities. The evaluated risk can be used to report potential areas of deficiency. If the lower level Software Life Cycle Process evaluation scores are entered, then another report page will be output containing those evaluation scores.

B.6.1.3 Report A3: Major Factor Percentile Chart. This report (see example in figure B-10) illustrates in a line graph the percentiles for each of the criteria and major factor evaluation scores relative to the historical evaluation data base. Scores above 75 percent are high, scores below 25 percent are low. Low scores may reflect deficiencies. The percentiles can be shown relative to all systems and relative to all systems of the same type as the system being evaluated. The example in figure B-10 is relative to systems of the same type.

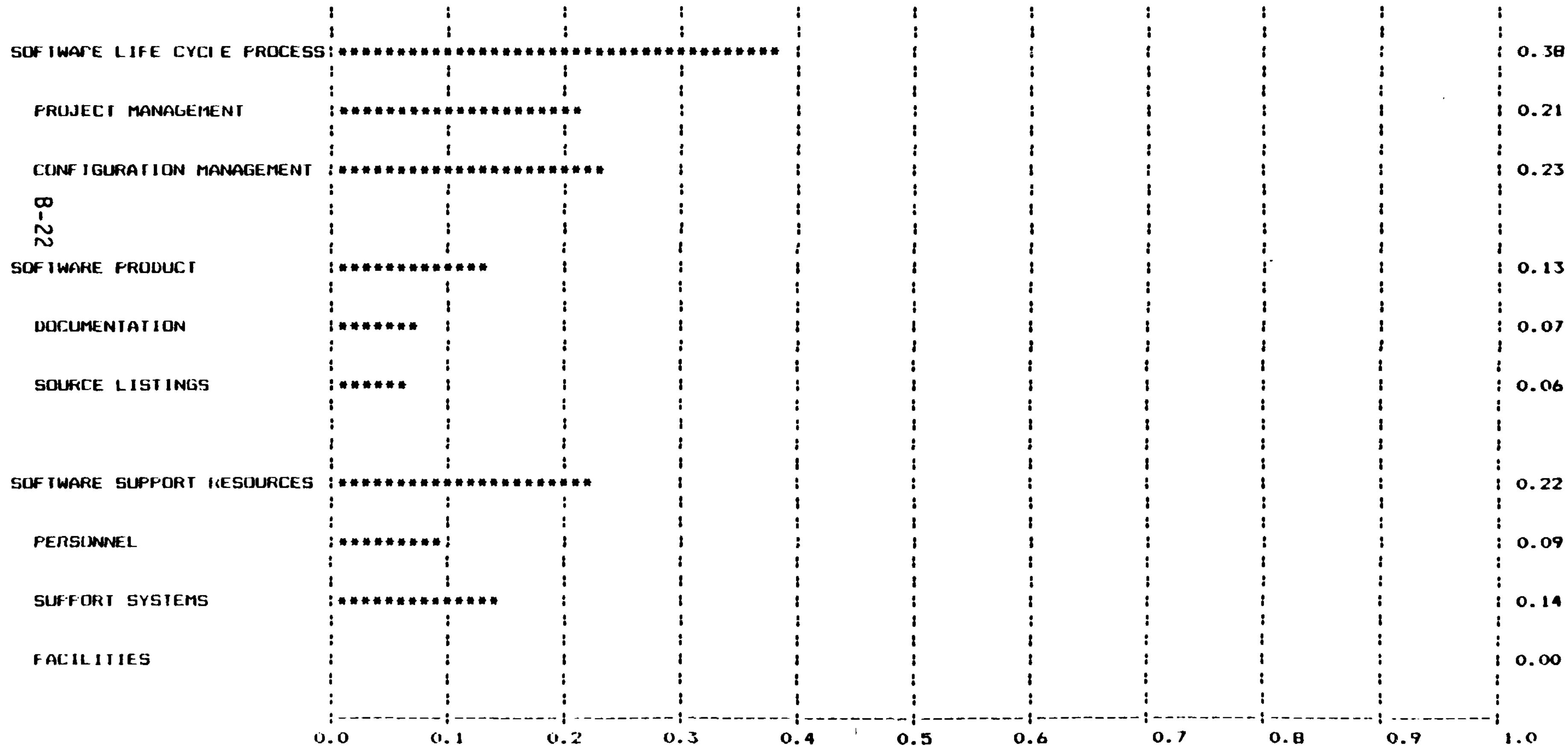
B.6.1.4 Report A4: Major Factor Risk Reduction Chart. This report (see example in figure B-11) illustrates in a line graph the maximum reduction in evaluated risk possible for each criteria and major factor. Those criteria/major factors which can effect large reductions in evaluated risk are termed risk drivers and are prime candidates for further analysis of potential risk reduction.

B.6.1.5 Report A5: Plot of Cumulative Distribution of Person Months Per Change Versus Risk. This report (see example in figure B-12) is

EVALUATION REPORT A4: MAJOR FACTOR RISK IMPACT CHART

SYSTEM PROFILE

SWSYSID :	SWTYPE : C-E
SYSTEM :	SUPPORTER: WR-ALC
SWSYSTEM :	USER : HQ-TAC



REDUCTION OF S/W SUPPORTABILITY RISK BY MAXIMUM FACTOR IMPROVEMENT

Figure B-11. Example Report A4: Major Factor Risk Impact Chart

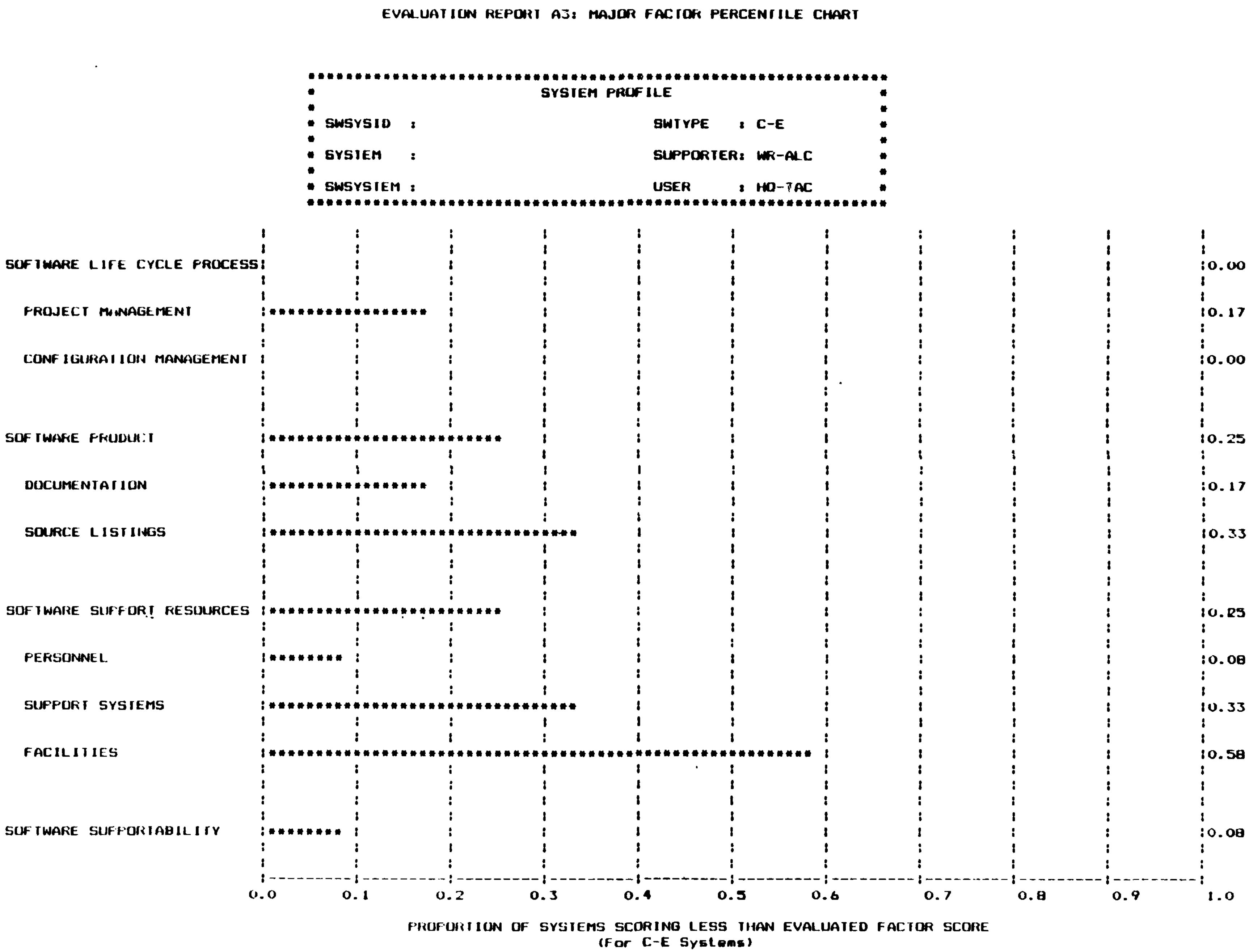
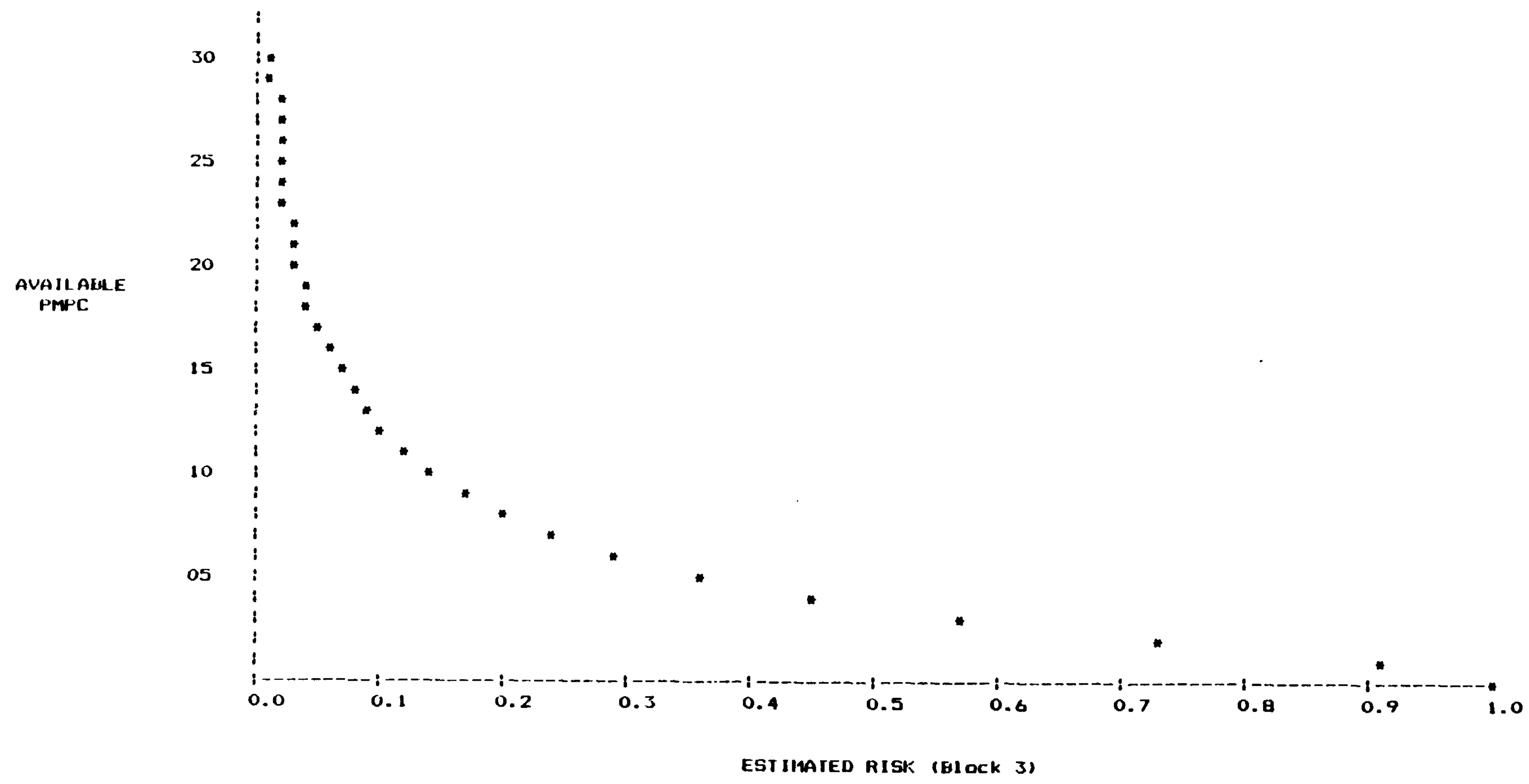


Figure B-10. Example Report A3: Major Factor Percentile Chart

EVALUATION REPORT A5: PLOT OF PMPC VS. RISK

SYSTEM PROFILE

SWSYSID :		SWTYPE :	C-E
SYSTEM :		SUPPORTER:	WR-ALC
SWSYSTEM :		USER :	HQ-TAC



Evaluated Risk: 0.55

Available PMPC: 3.29
Estimated PMPC: 3.57
Estimated Risk: 0.54

Figure B-12. Example Report A5: Plot of PMPC vs. Risk a plot showing the cumulative distribution and a table of evaluated and estimated risk and person months per change. The plot can be used for quick "what if" analysis of changes to the person months per change workload and/or the risk values. Plots are produced for data relative to each of the three block releases in the user/supporter baseline estimate.

B.6.1.6 Report A6: Summary of RAMSS Results. This report (see example in figure B-13) is a compact summary of information from the reports A1 through A5.

B.6.1.7 Report D1: Evaluation Data Base. This report (see the RAMSS User's Handbook for examples of all following reports) is a formatted table of all fields in the evaluation data base. This report is primarily used as a printed copy of the evaluation data.

B.6.1.8 Report D2: Maintenance Release Data Base. This report is a formatted table of all fields in the maintenance release data base. This report is primarily used as a printed copy of the evaluation data.

B.6.1.9 Report D3: Table of Evaluated Risk Regression Equation Coefficients. This report lists all coefficients used in the evaluated risk regression equation and the equations necessary to compute the evaluated risk.

B.6.1.10 Report D4: Table of Estimated Person Months Per Change Regression Equation Coefficients. This report lists all the coefficients used in the estimated person months per change regression equation and the equations necessary to compute the estimated person months per change.

```
               EVALUATION REPORT A6: SUMMARY OF SOFTWARE SUPPORTABILITY RISK ASSESSMENT

                    *************************************************************
                    *                       SYSTEM PROFILE                      *
                    *                                                           *
                    * SWSYSID  :                          SWTYPE    : C-E       *
                    *                                                           *
                    * SYSTEM   :                          SUPPORTER: WR-ALC     *
                    *                                                           *
                    * SWSYSTEM :                          USER      : HQ-TAC    *
                    *************************************************************

USER/SUPPORTER BASELINE ESTIMATE:

    SUPPORT STAFF:      15 Persons,  19% Dedicated, Avg.Skill level 3.00
                         9 Persons,  90% Dedicated, Avg.Skill level 3.00
    RELEASE SCHEDULE:  9.00 Month Block Release Cycle with  3.00 Month overlap
    BLOCK 1 CHANGE PROFILE: Total(  15); Type(  15,   0,   0); Complexity(   0,   0,  15); Priority(   0,   0,  15)
    BLOCK 2 CHANGE PROFILE: Total(  20); Type(  15,   5,   0); Complexity(   1,   4,  15); Priority(   1,   4,  15)
    BLOCK 3 CHANGE PROFILE: Total(  20); Type(  13,   7,   0); Complexity(   1,   6,  13); Priority(   1,   6,  13)

MAJOR FACTOR EVALUATION SCORES (THRESHOLD = 3.50, GOAL = 5.00):

    PROJECT MANAGEMENT:      3.47    LIFE CYCLE PROCESS: 3.32        SOFTWARE SUPPORTABILTY: 3.80
    CONFIGURATION MANAGEMENT 3.17

    DOCUMENTATION:           3.97    PRODUCT:            4.15
    SOURCE CODE:             4.32

    PERSONNEL:               3.53    SUPPORT RESOURCES   3.94
    SUPPORT SYSTEMS          3.72
    FACILITIES               4.58

MAJOR FACTOR RISK REDUCTION POTENTIAL:

    PROJECT MANAGEMENT:       0.21
    CONFIGURATION MANAGEMENT: 0.23

    DOCUMENTATION:            0.07
    SOURCE CODE:              0.06

    PERSONNEL:                0.09
    SUPPORT SYSTEMS:          0.14
    FACILITIES:               0.00

OVERALL RISK ASSESSMENT  (THRESHOLD = 0.50, GOAL = 0.20):

    ESTIMATED (Block 1): 0.14     LOW
              (Block 2): 0.48     MEDIUM
              (Block 3): 0.54     HIGH

    EVALUATED: 0.55     HIGH
```

Figure B-13. Example Report A6: Summary of Software Supportability Risk Assessment

B.6.1.11 Report D5: Table of AFOTEC Parameters (Threshold/Goal). This report lists the threshold and goal values for software supportability evaluation scores and the software supportability risk values. These threshold and goal values can be set by AFOTEC personnel and are only used to determine whether an evaluation score and/or risk value is considered to be HIGH, MEDIUM, or LOW.

B.6.2 BMDP Reports. There are seven possible BMDP reports which contain detailed statistical analysis data concerning the evaluation data, the evaluated risk regression model, the maintenance release data, and the estimated risk regression model. Data is passed to BMDP through ASCII files written by dBase III copy commands.

B.6.2.1 B1: Simple Data Description. This report lists all input evaluation data, various anomaly checks of the data, and univariate statistics.

B.6.2.2 B2: Histogram and Univariate Plots. This report provides a histogram and cumulative distribution plot of each major factor evaluation score.

B.6.2.3 B3: Multiple Linear Regression. This report provides the coefficients for the evaluated risk regression model. These coefficients must be manually entered into a dBase III file each time new or modified evaluation data is included in the regression analysis.

B.6.2.4 B4: Simple Data Description. This report is similar to B1, except it is for maintenance release data.

B.6.2.5 B5: Histogram and Univariate Plots. This report is similar to B2, except it is for maintenance release data.

B.6.2.6 B6: Description of Groups. This report provides histograms and analysis of variance information on certain stratified groups of maintenance release data variables.

B.6.2.7 B7: Multiple Linear Regression. This report is similar to B3, except it is for maintenance release data and the estimated person months per change. The standard error estimate as well as the regression coefficients must be manually entered into a dBase III file each time new or modified maintenance release data are included in the regression analysis.

B.7 ANALYZING RISK ASSESSMENT RESULTS.

The categorization of the risk assessment results as HIGH, MEDIUM, or LOW depends upon the values which distinguish the threshold and goal risk. The computation of the risk values depends upon the linear regression models. The analysis is primarily aided by the dBase III and BMDP printed reports.

B.7.1 Setting/Using Threshold and Goal Values

a. The recommended values for threshold and goal are:

(1) Software Supportability Evaluation
Goal: 5.0
Threshold: 3.5

(2) Software Supportability Risk
Goal: 0.20
Threshold: 0.5

(3) Software Supportability Percentiles
Goal: 75%
Threshold: 25%

These values are based upon experience and the current historial data base. They are somewhat subjective, and need to evolve over time. The values are used only as a reference in the summary report so as to distinguish scores which are HIGH, MEDIUM, or LOW.

b. The current threshold and goal values are set through a dBASE III screen if they need to be modified.

B.7.2 <u>The Estimated Risk Regression Equation</u>.

a. The estimated risk is determined from user/supporter baseline estimate parameters. First, a regression equation is used to determine the estimated workload in person months per change. Next, the available person months per change is computed from user/supporter baseline estimate parameters. Finally, the estimated risk is determined using a normal distribution of regression equation residuals with mean, the estimated person months per change, and standard deviation, the standard error of estimate of the regression equation. The estimated risk is the area under this normal curve above the available person months per change value. The regression equation for estimated person months is determined from the historical maintenance release data. The equations are as follows:

PMPC = person months per change

PMPC^ = estimated person months per change

$$PMPC\hat{} = e^{L\hat{}}$$

where

$$L\hat{} = b_0 + b_1\ (AVGSKILL) + b_2\ (PTCORR) + b_3\ (PCLOW)$$
$$+ b_4\ (PCHIGH) + b_5\ (PPNORM)$$
$$+ \sum_{i=\sigma}^{10} b_i\ (TYPE_i)$$

AVGSKILL = average skill (1-Low to 5-High) of support personnel

PTCORR = percentage of change requests which are corrections

PCLOW = percentage of change requests which are low complexity

PCHIGH = percentage of change requests which are high complexity

PPNORM = percentage of change requests which are normal priority

$TYPE_i$ = 1 if system type is same as $TYPE_i$

= 0 otherwise

The coefficients b_i are determined from a BMDP regression analysis program. These coefficients will change whenever the historical maintenance release data base is updated. Example coefficients are shown in figure B-11. The b_0 coefficient (INTERCEPT) incorporates the coefficient for the types SUP and SIM, so these types do not have coefficients specifically specified in figure B-14.

b. The available person months per change for a block release is determined from the user/supporter baseline estimate parameters for total number of personnel, total percentage dedicated to the software release (includes percentage dedicated to the software system and any release overlap), duration of the release cycle, and total number of changes in the release.

$PMPC_A$ = available person months per change

= (Number Persons * Percent Dedicated * Release Duration)/Number Changes

Estimated risk for an available PMPC, $PMPC_A$, is calculated as

$$\hat{R} = \text{Estimated Risk} = 1 - F\left[\frac{\ln(PMPC_A) - \hat{L}}{s}\right]$$

where

F(x) = cumulative standard normal distribution function evaluated at x

s = an estimate of the standard deviation of the normal distribution of the available person months per change about the $\hat{L}$.

```
PAGE   3  BMDP1R SW TYPE DUMMY VARIABLES PLUS COVARIATES
  REGRESSION TITLE IS
 SW TYPE DUMMY VARIABLES PLUS COVARIATES

0  DEPENDENT VARIABLE. . . . . . . . . . . . . . . . .      38 LN(PMFC)
   TOLERANCE . . . . . . . . . . . . . . . . . . . . .  0.0100
 ALL DATA CONSIDERED AS A SINGLE GROUP
0MULTIPLE R              0.5774          STD. ERROR OF EST.          0.9509
 MULTIPLE R-SQUARE       0.3334
```

ANALYSIS OF VARIANCE

	SUM OF SQUARES	DF	MEAN SQUARE	F RATIO	P(TAIL)
REGRESSION	63.3254	10	6.3325	7.003	0.0000
RESIDUAL	126.5911	140	0.9042		

VARIABLE		COEFFICIENT	STD. ERROR	STD. REG COEFF	T	P(2 TAIL)	TOLERANCE
INTERCEPT		0.07083					
AVGSKILL	7	0.54837	0.13224	0.323	4.147	0.0001	0.73516
PTCORR	29	-0.02948	0.39988	-0.007	-0.074	0.9413	0.52019
PCLOW	32	0.21815	0.40660	0.052	0.537	0.5924	0.51666
PCHIGH	34	2.09563	0.78857	0.255	2.658	0.0088	0.51551
PPNORM	35	-1.57228	0.61743	-0.208	-2.546	0.0120	0.71313
ATD	39	-0.50845	0.49983	-0.081	-1.017	0.3108	0.74867
ATE	40	0.14413	0.43891	0.033	0.328	0.7431	0.46024
C-E	41	0.35255	0.29391	0.133	1.200	0.2324	0.28386
EW	42	0.85802	0.46308	0.171	1.853	0.0660	0.55656
OFF	43	0.04114	0.29275	0.017	0.141	0.8884	0.33399

Figure B-14. Example Report: Baseline Estimate Risk Regression Analysis

The value of "s" is obtained from the BMDP-generated table of regression results ("STD. ERROR OF EST." in figure B-14); the function F may be numerically approximated by

$$F(x) = \frac{1}{2} + \left[SGN(x) \right] \left[\frac{1}{2} G \left(\frac{|x|}{\sqrt{2}} \right) \right]$$

$$G(z) = 1 - (1 + a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4)^{-4}$$

$a_1 = 0.278393$

$a_2 = 0.230389$

$a_3 = 0.000972$

$a_4 = 0.078108$

$$SGN(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases}$$

c. As an example from block 3 of the user/supporter baseline estimate in figure B-7, if

$$\begin{aligned} AVGSKILL &= 3.0 \\ PTCORR &= 0.65 \\ PCLOW &= 0.65 \\ PCHIGH &= 0.05 \\ PPNORM &= 0.65 \\ TYPE_{C-E} &= 1.0 \end{aligned}$$

then

$$\begin{aligned} \hat{L} = {} & 0.07083 + (0.54837)(3.0) + (-0.02948)(0.65) \\ & + (0.21815)(0.65) + (2.09568)(0.05) \\ & + (-1.57228)(0.65) + (0.35255)(1) \\ = {} & 1.2739275 \end{aligned}$$

and

$$\hat{PMPC} = 3.57$$

The available PMPC from the block 3 example of figure B-7 is computed as:

$$\begin{aligned} PMPC_A &= [(15*.19 + 9*.90)*(0.667)*9.0]/20 \\ &= 3.285 \end{aligned}$$

The estimated risk for block 3 of the example is therefore

$$\begin{aligned} \hat{R} &= 1 - F\left(\frac{\ell n(3.285) - 1.2739275}{0.9509}\right) \\ &= 1 - F(-0.0889272) \\ &= 1 - (0.5 + (-1)(0.5G(0.0628810))) \\ &= 1 - (0.5 - 0.5(1 - (1.0184180)^{-4})) \\ &= 1 - (0.5 - 0.5(0.0704010)) \\ &= 0.5 + 0.0352005 \\ &= 0.5352005 \end{aligned}$$

as shown in the example of figure B-7.

B.7.3 The Evaluated Risk Regression Equation

a. The evaluated risk is determined from the software supportability evaluation scores and a regression equation derived from the historical evaluation data. The equations are as follows:

$$R = \text{risk} = 1\text{–ACONFID} \qquad 0 \leq R \leq 1$$

$$R' = R(1 - a) + \frac{a}{2} \qquad a = 0.02$$

$$\hat{R} = \left[\frac{1}{1 + e^{-\hat{L}}} - \frac{a}{2}\right] \cdot (1 - a)^{-1}$$

$$\hat{L} = \ell n \frac{R'}{1 - R'}$$

$$b_0 + b_1 \text{(APRODUCT)} + b_2 \text{(AEPER)} + b_3 \text{(AESYS)} + b_4 \text{(AEFAC)} + b_5 \text{(AMANAGE)}$$

APRODUCT = Software Product Maintainability evaluation score
AEPER = Support Resources Personnel evaluation score
AESYS = Support Resources System evaluation score
AEFAC = Support Resources Facilities evaluation score
AMANAGE = Software Life Cycle Process evaluation score

The coefficients b_i are determined from a BMDP regression analysis program. These coefficients will change whenever the historical evaluation data base is updated. Example coefficients are shown in figure B-15.

```
PAGE   3  BMDP1R L(RISK) VS. SUPPORTABILITY FACTORS
  REGRESSION TITLE IS
L(RISK) VS. SUPPORTABILITY FACTORS

0  DEPENDENT VARIABLE. . . . . . . . . . . . . . . .       15 LRISK
   TOLERANCE . . . . . . . . . . . . . . . . . . . .   0.0100
ALL DATA CONSIDERED AS A SINGLE GROUP
0MULTIPLE R            0.8086        STD. ERROR OF EST.         0.6239
MULTIPLE R-SQUARE      0.6539
```

ANALYSIS OF VARIANCE

	SUM OF SQUARES	DF	MEAN SQUARE	F RATIO	P(TAIL)
REGRESSION	38.2376	5	7.6475	19.650	0.0000
RESIDUAL	20.2382	52	0.3892		

VARIABLE		COEFFICIENT	STD. ERROR	STD. REG COEFF	T	P(2 TAIL)	TOLERANCE
INTERCEPT		4.90401					
APRODUCT	4	-0.29131	0.13090	-0.226	-2.225	0.0304	0.64641
AEPER	5	-0.15600	0.12403	-0.136	-1.258	0.2141	0.56613
AESYS	6	-0.25120	0.12905	-0.181	-1.946	0.0570	0.77162
AEFAC	7	0.04294	0.10015	0.043	0.429	0.6699	0.66939
AMANAGE	11	-0.66174	0.14256	-0.509	-4.642	0.0000	0.55331

Figure B-15. Example Report: Evaluation Risk Regression Analysis

b. As an example from the evaluation scores illustrated in figure B-9, if

APRODUCT = 4.15

AEPER = 3.53

AESYS = 3.72

AEFAC = 4.58

AMANAGE = 3.32

then

$$\hat{L} = 4.90401 + (-0.29131)(4.15) + (-0.15600)(3.53) + (-0.25120)(3.72) + (0.04294)(4.58) + (-0.66174)(3.32)$$

$$= 0.20962$$

and

$$\hat{R} = \left[\frac{1}{1 + e^{-0.20962}} - \frac{.02}{2}\right] (1-.02)^{-1}$$

$$= 0.55$$

B.7.4 Analyzing Results of the Assessment Reports

a. There are six risk assessment reports output from the RAMSS automated support system:

(1) A1: User/Supporter Baseline Estimate

(2) A2: Table of Evaluation Scores

(3) A3: Major Factor Percentile Chart

(4) A4: Major Factor Risk Reduction Chart

(5) A5: Plot of Cumulative Distribution of Person Months Per

(6) A6: Summary of RAMSS Results

b. An example of the report A1 is shown in figure B-7. Examples of reports A2, A3, A4, A5, and A6 are shown in figures B-9 through B-13, respectively. A brief explanation of these reports is contained in section B.6. Detailed explanations of these reports are contained in the RAMSS User's Handbook.

c. For the data in the example reports the following analysis conclusions hold:

(1) From report A6: the evaluated software supportability risk is HIGH; the estimated software supportability risk is MEDIUM; the main risk driver is software life cycle process; support resources personnel had a somewhat low evaluation score but there was not much potential for reduction of risk by improvements in this characteristic

(2) From report A4: the amount of risk reduction is greatest for software configuration management; this major factor is the prime candidate for potential improvement to reduce software supportability risk

(3) From report A3: there is support for the conclusion that the overall evaluation is LOW relative to the systems in the evaluation data base.

Suppose the evaluator were able to analyze the detailed evaluation results and conclude that the overall software life cycle process score could be improved from 3.32 to 4.58. The corresponding reduction in evaluated risk would be from 0.55 to 0.35, and the various analysis reports would reflect that overall improvement in the software life cycle process evaluation results.

B.8 REPORTING RISK ASSESSMENT RESULTS.

The risk assessment results which should be reported include:

(1) Evaluated software supportability risk (report A2)

(2) Estimated software supportability risk (report A1)

(3) Major risk drivers (report A4)

(4) Risk reduction potential (report A4)

(5) Individual characteristics anomalies (all reports)

- a) Above risk threshold (0.50)
- b) Below percentile threshold (0.25)
- c) Below evaluation threshold (3.50)
- d) Goal or better characteristics (0.20, 0.75, 5.0)

An overall assessment of the software supportability risk as HIGH, MEDIUM, or LOW on the basis of the matrix in figure B-16 should be reported.

RISK ASSESSMENT \ RISK AREA	ESTIMATED RISK (BLOCK 1)	ESTIMATED RISK (BLOCK 2)	ESTIMATED RISK (BLOCK 3)	EVALUATED RISK	PERCENTILE SCORE (OVERALL)
LOW					
MEDIUM					
HIGH					

86-0360-TR-G-05

Figure B-16. Software Supportability Risk Assessment Matrix

B.9 SUMMARY OF THE RAMSS PHILOSOPHY.

a. The following is a summary of the general philosophy which should guide the evaluator (e.g., STM/DSE) in conducting a RAMSS.

b. The evaluator should understand that the RAMSS is not yet mature. It is critical that the evaluator review all aspects of the risk in order to arrive at an overall assessment of a software's supportability risk to the Air Force.

c. The evaluator should always be prepared to describe why the evaluated or estimated risk is HIGH or LOW by tracing to specific criteria, major factors, characteristics, resource workload, or baseline change profile for supporting data.

d. The evaluator should be very familiar with the RAMSS automated support system, or there should be an AFOTEC support person who is familiar with it and can assist the evaluator.

e. The historical data for evaluations and maintenance releases are immature. Care should be exercised in relying too heavily on these data. These data need to be improved over time. Anomalies in results may be because of incomplete data.

ATTACHMENT B1

GLOSSARY OF TERMS

B.1.1 INTRODUCTION.

a. The glossary of terms for the RAMSS has varied as the methodology development has progressed. Refer to BDM/A-84-322-TR (Final) dated September 28, 1984, for a complete glossary of terms relating to risk assessment.

b. Some terms have more than one description; when this is the case, the description either:

(1) Are significantly different between sources (though the effective meaning may be not much different)

(2) Are used differently (different users or technical language)

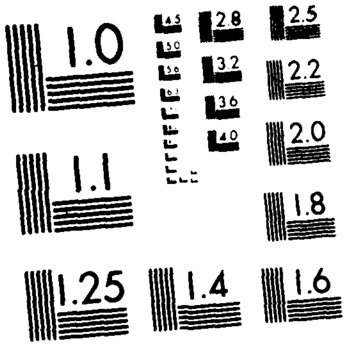
(3) May be found within the context of a different source

(4) Have real differences in meaning.

Both DoD and non-DoD (e.g., FIPS PUBs, NBS Special Publications) sources are used. The non-DoD sources and terms are not mandated for our use, but are rather included for breadth of understanding, for those relevant terms commonly used with the non-DoD governmental and/or private sectors.

c. The source of each description is indicated by a symbol in parentheses before that source's term description:

AD-A191 075 RISK ASSESSMENT METHODOLOGY FOR SOFTWARE SUPPORTABILITY 3/3
(RAMSS): PILOT EV..(U) BDM CORP ALBUQUERQUE NM
D E PEERCY ET AL. 14 APR 86 BDM/ABQ-86-0360-TR
UNCLASSIFIED F29601-85-C-0058 F/G 12/5 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

$TERM_1$

(SYMBOL$_{1.1}$)

Description$_{1.1}$...

(SYMBOL$_{1.2}$)

Description$_{1.2}$...

.

.

.

(SYMBOL$_{1.n}$)

Description$_{1.n}$...

$TERM_2$

.

.

.

$TERM_N$

The symbols used and corresponding sources are:

(AFOTECP1)	AFOTECP 800-2, Volume I, 10 Nov 82, "Software Test Manager's Guide."
(AFOTECP3)	AFOTECP 800-2, Volume III, 1 Jan 84, "Software Maintainability Evaluator's Guide."
(AFOTECP5)	AFOTEC 800-2, Volume V, 25 Jul 83, "Software Support Facility Evaluation--User's Guide."
(AFR55-43)	Air Force Regulation 55-43, "Management of Operational Test and Evaluation", 28 Jun 1985.
(AFR800-14)	Air Force Regulation 800-14, Volume I, "Management of Computer Resources in Systems," 12 Sep 75.
(DoD480A)	DoD Standard 480A, "Configuration Control - Engineering Changes, Deviations and Waivers", 12 Apr 78.
(ROWE)	Rowe, William, <u>An Anatomy of Risk</u>, John Wiley, 1977.
(CURRENT)	Current document definition.

B.1.2 GLOSSARY OF TERMS FOR DEVELOPING AND IMPLEMENTING A RISK ASSESSMENT METHODOLOGY FOR SOFTWARE SUPPORTABILITY.

Allocated Baseline

(DoD480A)
See Baseline.

Allocated Configuration Identification

(DoD480A)
Current, approved performance oriented specifications governing the development of configuration items that are part of a higher level CI, in which each specification (1) defines the functional characteristics that are allocated from those of the higher level CI, (2) establishes the tests required to demonstrate achievement of its allocated functional characteristics, (3) delineates necessary interface requirements with other associated configuration items, and (4) establishes design constraints, if any, such as component standardization, use of inventory items, and integrated logistic support requirements.

Application Software

(AFOTECP5)
The software written by software support personnel, or purchased from a contractor, used directly in supporting ECSs. It is normally used for simulation, testing, and ECS code development.

Automated Software Development Tool

(AFOTECP5)
A component of System Software that assists in the design, implementation, documentation, and verification of ECS software.

Availability

(AFR800-14)
A measure of the degree to which an item is in the operable and commitable state at the start of the mission, when the mission is called for at an unknown (random) point in time. (MIL-STD-721)

(AFOTECP5)
The probability that a system is operating satisfactorily at any point in time when used under stated conditions.

Available Person Time (APT)

(CURRENT)
The software support person-months available for a particular software release computed as the product of the release duration

in months, the number of support personnel, and the percentage of the time those personnel are dedicated to the subject software release (versus shared across other releases or other software systems). This time includes overhead activity directly related to the subject release. The release duration is the release engineering completion date minus the release start date.

Baseline

(DoD480A)
A configuration identification document or a set of such documents formally designated and fixed at a specific time during a CI's life cycle. Baselines, plus approved changes from those baselines, constitute the current configuration identification. For configuration management there are three baselines, as follows:

a) Functional Baseline. The initial approved functional configuration identification.

b) Allocated Baseline. The initial approved allocated configuration identification.

c) Product Baseline. The initial approved or conditionally approved product configuration identification.

(ROWE)
A known reference used as a guide for further development activities.

Baseline Profile

(CURRENT)
See Baseline Software Change Profile.

Baseline Software Change Profile

(CURRENT)
The set of numbers (or any subset) determined by specifying the number of requests per release for each request category. A request category is the triple (type, priority, complexity) where type is conversion, enhancement, or correction; priority is emergency, urgent, or normal; and complexity is high, medium, low.

Baseline Software Supportability Estimate

(CURRENT)
See User/Supporter Baseline Estimate

Block Release

(CURRENT)
See Release.

Change Control

(DoD480A)
See Configuration Control

Complexity of MA

(CURRENT)
See Maintenance Complexity

Computer Program

(AFR800-14)
A series of instructions or statements in a form acceptable to an electronic computer, designed to cause the computer to execute an operation or operations.

Computer Program Configuration Item (CPCI)

(CURRENT)
See Computer Software Configuration Item

Computer Resources

(CURRENT)
The totality of computer hardware, computer software, personnel, documentation, supplies, and services.

(AFR800-14)
The totality of computer equipment, computer programs, associated documentation, contractual services, personnel and supplies.

Computer Resources Integrated Support Plan (CRISP)

(AFR55-33)
The CRISP identifies organizational relationships and responsibilities for the management and technical support of computer resources. It functions during the full-scale development (FSD) phase to identify computer resources necessary to support computer programs after program management responsibility and system turnover are transferred. After the transfer, the CRISP continues to function as the basic agreement between the supporting and using commands for management and support of computer resources.

Computer Resources Working Group (CRWG)

(CURRENT)
A group comprised of all the participating commands (for a particular system) which writes and updates the Computer Resources Integrated Support Plan (CRISP). The group insures that necessary elements of the CRISP are included in transfer and turnover agreements.

Computer Software Configuration Item (CSCI)

(CURRENT)
See Configuration Item

Configuration Audit

(CURRENT)
The process of verifying that all required configuration items have been produced, that the current version agrees with specified requirements, that the technical documentation completely and accurately describes the configuration items, and that all change requests have been resolved.

Configuration Control

(DoD480A)
The systematic evaluation, coordination, approval or disapproval, and implementation of all approved changes in the configuration of a configuration item after formal establishment of its configuration identification.

Configuration Identification

(DoD480A)
The current approved or conditionally approved technical documentation for a configuration item as set forth in specifications, drawings and associated lists, and documents referenced therein.

Configuration Index

(CURRENT)
This document, produced by the development contractor, reports the current status of configuration item development in terms of specifications and other documents that depend on the configuration, such as qualification Test Plans and Procedures, User Manuals, and the Version Description Document. It lists all ECPs and SCNs incorporated, approved ECPs not yet incorporated, and other data.

Configuration Item (CI)

(AFR800-14)
An aggregation of equipment/software, or any of its discrete portions, which satisfies an end use function and is designated by the Government for configuration management. CIs may vary widely in complexity, size and type, from an aircraft or electronic system to a test meter or round of ammunition. During development and initial production, CIs are only those specification items that are referenced directly in a contract (or an equivalent in-house agreement). During the operation and maintenance period, any repairable item designated for separate procurement is a configuration item (AFR 65-3).

Configuration Management (CM)

(DoD480A)
A discipline applying technical and administrative direction and surveillance to (1) identify and document the functional and physical characteristics of a configuration item, (2) control changes to those characteristics, and (3) record and report change processing and implementation status.

Configuration Management Plan (CMP)

(CURRENT)
A document which describes project responsibilities and procedures for implementing CM.

Configuration Management System (CMS)

(AFOTECP5)
A system applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configuration item; to control changes to those characteristics and to record and report change processing and implementation status.

Configuration Status Accounting

(DoD480A)
The recording and reporting of the information that is needed to manage a configuration effectively, including a listing of the approved configuration identification, the status of proposed changes to the configuration, and the implementation status of approved changes.

Consistency

(CURRENT)
A measure of the extent the software products correlate and contain uniform notation, terminology, and symbology.

Conversion (Adaptive) MA

(CURRENT)
See Maintenance Type.

Corrective MA

(CURRENT)
See Maintenance Type.

Critical Issues

(AFOTECP1)
Those aspects of a system's capability, either operational, technical, or other, that must be questioned before a system's overall worth can be estimated and that are of primary importance to the decision authority in reaching a decision to allow the system to advance into the next acquisition phase (DoD Directive 5000.3).

Data Item Description

(AFR800-14)
A form which specifies an item of data required to be furnished by a contractor. This form specifically defines the content, preparation instructions, format and intended use of each data product.

Descriptiveness

(CURRENT)
A measure of the extent that software products contain information regarding its objectives, assumptions, inputs, processing, outputs, components, revision status, etc.

Development Contractor Activity

(CURRENT)
Those organizations responsible for development of a system in order to achieve an initial operational capability. Organizations include the prime development contractor and any subcontractors to the prime contractor.

Documentation

(AFOTECP5)
All of the written work describing operating and maintenance procedures for a system.

Embedded Computer Resources

(AFOTECP1)
Computer resources incorporated as integral parts of, dedicated to, required for direct support of, or for the upgrading or

modification of major or less than major system(s) (excludes ADP resources as defined and administered under AFR 300 series) (USAF/RD/LE Policy letter, 13 October 1981).

Embedded Computer System (ECS)

(AFOTECP1)

a) A computer that is integral to an electromechanical system and that has the following key attributes:

   (1) Physically incorporated into a large system whose primary function is not data processing

   (2) Integral to, or supportive of, a larger system from a design, procurement, and operations viewpoint

   (3) Inputs include target data, environmental data, command and control, etc.

   (4) Outputs include target information, flight information, control signals, etc.

b) In general, an embedded computer system (ECS) is developed, acquired, and operated under decentralized management (DoD Directives 5000.1, 5000.2).

Emergency MA

(CURRENT)
See Maintenance Priority.

Engineering Change Proposal (ECP)

(AFR55-43)
A formal, priced document (DD Form 1692) used to propose changes to the contact provisions and scope, if not partially waived (see Contract Change Proposal), and to the configuration item baseline identification especially when related equipment, critical issues, interfaces, or technical manuals are affected or retrofit is involved. See MIL-STDs 480, 481, and 483; and AFR 400-3.

Enhancement (Perfective) MA

(CURRENT)
See Maintenance Type.

Estimated Person Months Per Change

(CURRENT)
See Person Months Per Change

Estimated Risk

(CURRENT)
See Software Supportability Risk

Estimation

(ROWE)
The assignment of probability measures to a postulated future event.

Evaluated Person Months Per Change

(CURRENT)
See Person Months Per Change

Evaluated Risk

(CURRENT)
See Software Supportability Risk.

Evaluation

(ROWE)
Comparison of an activity performance with the objectives of the activity and assignment of a success measure to that performance.

Evaluation Criteria

(AFOTECP1)
Standards by which achievement of required operational effectiveness/suitability characteristics or resolution of technical or operational issues may be judged. For full-scale development and beyond, evaluation criteria must include quantitative goals (the desired value) and thresholds (the value beyond which the characteristic is unsatisfactory) whenever possible (DoD Directive 5000.3).

Expandability

(CURRENT)
A measure of the extent that a physical change to information, computational functions, data storage, or execution time can be easily accomplished once the nature of what is to be changed is understood.

(AFOTECP5)
A measure of the ease with which the functional capability of computer hardware or software may be expanded.

Facility

(AFOTECP5)
The physical plant and the services it provides; specific examples are physical space, electrical power, physical and electromagnetic (TEMPEST) security, environmental control, fire safety provisions, and communications availability.

Firmware

(AFOTECP1)
a) Computer programs and data loaded in a class of memory that cannot be dynamically modified by the computer during processing.
b) Hardware that contains a computer program and data that cannot be changed in its application environment.

Note 1. Computer programs and data contained in firmware are classified as software; the circuitry containing the computer program and data is classified as hardware (Data and Analysis Center for Software).

Functional Baseline

(DoD480A)
See Baseline.

Functional Configuration Audit (FCA)

(DoD480A)
The formal examination of functional characteristics test data for a configuration item, prior to acceptance, to verify that the item has achieved the performance specified in its functional or allocated configuration identification.

Functional Configuration Identification

(DoD480A)
The current approved technical documentation for a configuration item which prescribes (1) all necessary functional characteristics, (2) the tests required to demonstrate achievement of specified functional characteristics, (3) the necessary interface characteristics with associated CI's, (4) the CI's key functional characteristics and its key lower level CI's, if any, and (5) design constraints, such as envelope dimensions, component standardization, use of inventory items, integrated logistics support policies.

High Complexity MA

(CURRENT)
See Maintenance Complexity.

Historical Maintenance Profile

(CURRENT)
A histogram of data on software system releases, with the x-axis representing discrete ranges of (available) person-months per change and the y-axis representing the number of software system releases that fall into each x-axis discrete range. For purposes of analysis or illustration, the axes may be reversed.

Independent Verification and Validation (IV&V)

(AFOTECP1)
An independent assessment process structured to ensure that computer programs fulfill the requirements stated in system and subsystem specifications and satisfactorily perform the functions required to meet the user's and supporter's requirements. IV&V consists of three essential elements: independent, verification, and validation:

(1) Independent. An organization/agency which is separate from the software development activity from a contractual and organizational standpoint.

(2) Verification. The evaluation to determine whether the products of each step of the computer program development process fulfill all requirements levied by the previous step.

(3) Validation. The integration, testing, and/or evaluation activities carried out at the system/subsystem level to evaluate the developed computer program against the system specifications and the user's and supporter's requirements (AFR 88-14).

Initial Operational Capability (IOC)

(CURRENT)
That point in a system's life cycle when the agreed upon number of production systems has been delivered to the user (using command) for operational use.

Instrumentation

(CURRENT)
A measure of the extent that software products contain aids that enhance testing.

Interface Control Working Group (ICWG)

(MIL-STD-483)
For programs which encompass a system/HWCI/CSCI design cycle, an ICWG normally is established to control interface activity between contractors or agencies, including resolution of interface problems and documentation of interface agreements.

Interoperability

(AFOTECP5)
A measure of the degree to which computer hardware/software can interface to and operate with other similar computer hardware/software.

Low Complexity MA

(CURRENT)
See Maintenance Complexity.

Maintainability

(AFOTECP5)
The probability that a system out of service for maintenance can be properly repaired and returned to service in a stated elapsed time.

Maintenance Complexity

(CURRENT)
The general degree of difficulty to complete a maintenance request: high, medium, low.

High: An MA where changes are in requirements, design, code, and test; or greater than 10 percent of CSCI is affected; or several modules are affected by the change (global changes); or the technical nature of the change requires highly specialized personnel skills; or the level of effort by personnel is large.

Medium: An MA where changes are in design, code and test; or between 1 percent and 10 percent of CSCI is affected; or at least two modules are affected by the change (semi-local); or the level of effort by personnel is average.

Low: An MA where changes are isolated to only one unit (e.g., one module/compilation unit) of code; or no more than 1 percent of CSCI is affected; or the level of effort by personnel is minimal.

Maintenance Documentation

(AFOTECP5)
The documentation that describes the maintenance of computer system hardware and software.

Maintenance Priority

(CURRENT)
The criticality of the maintenance request in order to preserve mission readiness; emergency, urgent, normal.

Emergency; An MA requiring all available personnel's dedicated effort to correct the problem as soon as possible (e.g., 24 hours); MIL-STD-1679 severity code 1 or 2: mission termination or severe degradation.

Urgent: An MA requiring next "block release" turnaround; MIL-STD-1679 severity code 3: mission impact.

Normal: An MA not in the Emergency or Urgent categories; MIL-STD-1679 severity code 4 or 5: mission inconvenience.

Maintenance Profile

(CURRENT)
See Historical Maintenance Profile.

Maintenance Request Category

(CURRENT)
The identification of a maintenance request by specification of the maintenance priority, type, and complexity.

Maintenance Type

(CURRENT)
The type of maintenance actions required to complete a maintenance request: conversion, enhancement, correction.

Conversion (Adaptive) MA: Any change/effort to a software system which is initiated as a result of changes in the environment (e.g., hardware, system software) in which the software system must operate.

Enhancement (Perfective) MA: Any change, insertion, deletion, modification, or extension made to a software system to meet the evolving needs of the user.

Corrective MA: Any change which is necessitated by actual faults (induced or residual) in a software system.

Medium Complexity MA

(CURRENT)
See Maintenance Complexity.

Modularity

(CURRENT)
A measure of the extent that a logical partitioning of software products into parts, components, and/or modules has occurred.

Normal MA

(CURRENT)
See Maintenance Priority.

Operation Support Activity

(CURRENT)
Those organizations responsible for post deployment operation and support of a system. Organizations include the using command, supporting command, contractors (if used), and test and evaluation agencies (if used).

Operational Effectiveness

(AFOTECP1)
The overall degree of mission accomplishment of a system used by representative personnel in the context of the organization, doctrine, tactics, threat (including countermeasures and nuclear threats), and environment in the planned operational employment of the system (DoD Directive 5000.3).

Operational Suitability

(AFOTECP1)
The degree to which a system can be satisfactorily placed in field use, with consideration being given to availability, compatibility, transportability, interoperability, reliability, wartime usage rates, maintainability, safety, human factors, manpower supportability, logistic supportability, and training requirements (DoD Directive 5000.3).

Person-Months per Change (PMPC)

(CURRENT)
Available PMPC: Raw personnel resources workload to support a user/supporter baseline workload estimate of a specified number of changes. Computed as the number of full-time equivalent personnel times the release cycle in months divided by the total number of changes.

Estimated PMPC: An estimate of a personnel resources workload required to support the user/supporter baseline estimate. This estimate is computed by using a regression equation whose coefficients are derived from historical maintenance release data.

Evaluated PMPC: A realistic estimate of personnel resources workload effectiveness to support the user/supporter baseline estimate as derived from an evaluation of the software supportability characteristics.

Personnel

(CURRENT)
See Support Personnel.

Personnel Skill Level

(CURRENT)
A subjective integer rating from 1 (lowest) to 5 (highest) of software support personnel experience, education, and specific task responsibility capabilities.

Physical Configuration Audit (PCA)

(DoD480A)
The formal examination of the "as-built" configuration of a unit of a CI against its technical documentation in order to establish the CI's initial product configuration identification.

Priority

(CURRENT)
See Maintenance Priority.

Probability

(ROWE)
A numerical property attached to an activity or event whereby the likelihood of its future occurrence is expressed or clarified.

Probability Distribution

(ROWE)
The representation of a repeatable stochastic process by a function satisfying the axioms of probability theory.

Probability of Occurrence

(ROWE)
The probability that a particular event will occur, or will occur in a given interval.

Procurement Activity

(CURRENT)
Those government organizations responsible for assuring delivery of a production system. Organizations include the program office, implementing command, development and operational test and evaluation agencies, and appropriate independent verification and validation agencies if used.

Product Baseline

(DoD480A)
See Baseline.

Product Configuration Identification

(DoD480A)
The current approved or conditionally approved technical documentation which defines the configuration of a CI during the production, operation, maintenance, and logistics support phases of its life cycle, and which prescribes (1) all necessary physical or form, fit and function characteristics of a CI, (2) the selected functional characteristics designated for production acceptance testing, and (3) the production acceptance tests.

Program Management Directive (PMD)

(AFR800-14)
The official HQ USAF management directive used to provide direction to the implementing and participating commands and satisfy documentation requirements. It will be used during the entire acquisition cycle to state requirements and request studies as well as initiate, approve, change, transition, modify or terminate programs. The content of the PMD, including the required HQ USAF review and approval actions, is tailored to the needs of each individual program (AFR 800-2).

Program Management Plan (PMP)

(AFR800-14)
The document developed and issued by the Program Manager which shows the integrated time-phased tasks and resources required to complete the task specified in the PMD. The PMP is tailored to the needs of each individual program (AFR 800-2).

Program Management Responsibility Transfer (PMRT)

(AFR800-14)
That point in time when the designated Supporting Command accepts program management responsibilities from the Implementing Command. This includes logistic support and related engineering and procurement responsibilities (AFR 800-4).

Program Support Tools

(AFOTECP3)
General debug aids, test/retest software, trace software/hardware features, use of compiler/link editor, library management/configuration management/text editor/display software tools.

Program Test Plan

(AFOTECP3)
Set of descriptions and procedures for how the program is to be (or can be, or has been) tested.

Quality Assurance (QA)

(CURRENT)
All actions that are taken to assure that a development organization delivers products that meet performance requirements and adhere to standards and procedures.

Release

(CURRENT)
A version of a software system representing either the initial baseline configuration or an update to a previous version that incorporates a defined set of software change requests. Each release becomes a new baseline configuration.

Release Engineering Completion Data

(CURRENT)
The date when the software engineering activity for a release is complete. The software engineering activity includes configuration management, quality assurance, and software maintenance project phases of requirements, design, code, unit test, integration test, and operational test. Activity including "kit proofing," prom burning, and in general technical order modifications which typically occur between engineering completion and field implementation (distribution) is <u>not</u> included.

Release Field Date

(CURRENT)
The date when a software system release is officially distributed and implemented in the field for operational use.

Release ID

(CURRENT)
A unique identifier for a software system release.

Release Start Date

(CURRENT)
The date when major analysis activity related to a specified release begins for which software support resources are required.

Reliability

(ROWE)
The probability that the system will perform its required functions under given conditions for a specified operating time.

Risk

(ROWE)
The potential for realization of unwanted, negative consequences of an event.

Risk Acceptance

(ROWE)
Willingness of an individual, group, or society to accept a specific level of risk to obtain some gain or benefit.

Risk Acceptance Function

(ROWE)
A subjective operator relating the levels of probability of occurrence and value of a consequence to a level of risk acceptance.

Risk Acceptance Level

(ROWE)
The acceptable probability of occurrence of a specific consequence value to a given risk agent.

Risk Acceptance Utility Function

(ROWE)
The profile of the acceptability of the probability of occurrence for all consequences involved in a risk situation for a specific risk agent.

Risk Agent

(ROWE)
A person or group of persons who evaluates directly the consequences of a risk to which the person or group of persons is subjected.

Risk Assessment

(ROWE)
The total process of quantifying a risk and finding an acceptable level of that risk for an individual, group, or society. It involves both risk determination and risk evaluation.

Risk Assessment Methodology for Software Supportability (RAMSS)

(CURRENT)
A method of determining the disparity between the estimated risk (determined from the support concept, baseline software supportability profile, and historical maintenance profile) and the evaluated risk (determined from a conversion of the software supportability evaluation metrics).

Risk Consequence

(ROWE)
The impact to a risk agent of exposure to a risky event.

Risk Determination

(ROWE)
The process of identifying and estimating the magnitude of risk.

Risk Estimation

(ROWE)
The process of quantification of the probabilities and consequence values for an identified risk.

Risk Evaluation

(ROWE)
The complex process of developing acceptable levels of risk to individuals or society.

Risk Profile Baseline

(CURRENT)
The measure of information and/or requirements which serve as the zero reference against which negative (and positive) outcomes can be determined.

Risk Reduction

(ROWE)
The action of lowering the probability of occurrence and/or the value of a risk consequence, thereby reducing the magnitude of the risk.

Sensitivity Analysis

(ROWE)
A method used to examine the operation of a system by measuring the deviation of its nominal behavior due to perturbations in the performance of its components from their nominal values.

Simplicity

(CURRENT)
A measure of the extent that software products reflect the use of singularity concepts and fundamental structures in organization, language, and implementation techniques.

Simulation

(AFR800-14)
The representation of physical systems or phenomena by computers, models or other equipment.

Site

(CURRENT)
A software support site, or particular location, where software support activity is being accomplished. Includes sites such as the Air Logistics Centers (ALCs).

Site Survey Form

(CURRENT)
The data collection form used during the software support site visits to collect background, evaluation, and maintenance release data.

Software

(AFOTECP1)
A set of computer programs, procedures, and associated documentation concerned with the operation of a data processing system.

(CURRENT)
The programs which execute in a computer. The data input, output, and controls upon which program execution depends and the documentation which describes, in a textual medium, development and maintenance of the program.

Software Change Request

(CURRENT)
An official request that could involve a change to a software system. Such requests include problem report, enhancement

requirement, modification request, or any other form that is officially tracked by a configuration management function.

Software Configuration Management

(CURRENT)
A discipline applying technical and administrative direction and surveillance to 1) identify and document the functional and physical characteristics of a configuration item, 2) control changes to those characteristics, and 3) record and report change processing and implementation status.

Software Delivery

(CURRENT)
That point in the software life cycle when the software support function assumes responsibility for the "next" set of configuration changes to the software (e.g., next block release). This point is logically no later than PMRT, but could be as early as IOC. This applies when a contractor or government agency assumes the software support function.

Software Error

(CURRENT)
The human decision (inadvertent or by design) which results in the inclusion of a fault in a software product.

Software Fault

(CURRENT)
The presence or absence of that part of a software product which can result in software failure.

Software Life Cycle Process

(CURRENT)
The policy, methodology, procedures, and guidelines applied in a software environment to the software development and support life cycle activities.

Software Maintainability

(AFOTECP1)
The ease with which software can be changed in order to:
(1) Correct errors
(2) Add/modify system capabilities through software changes
(3) Delete features from programs
(4) Modify software to be compatible with hardware changes.

(CURRENT)
A quality of software which reflects the effort required to perform software maintenance actions.

Software Maintenance

(CURRENT)
Those actions required for:
(1) Correction - Removal, correction of software faults
(2) Enhancement - Addition/deletion of features from the software
(3) Conversion - Modification of the software because of environment (data hardware) changes.

Software Maintenance Environment

(CURRENT)
An integration of personnel support systems and physical facilities for the purpose of maintaining software products.

Software Maintenance Measures

(CURRENT)
Measures of software maintainability and environment capabilities to support software maintenance activity.

Software Maintenance Project Management

(CURRENT)
The software life cycle process management applied during the support phase for the software to accomplish specific software maintenance tasks which derive from software problem reports or change requests.

Software Management

(CURRENT)
The policy, methodology, procedures, and guidelines applied in a software environment to the software development/maintenance activities. Also, those personnel with software management responsibilities.

Software Project Management

(CURRENT)
See Software Management.

Software Project Management Design Methods

(CURRENT)
The software project management process utilizes design methods which enhance software supportability to the extent that design

methodology standards and conventions are: 1) documented, followed, and validated through quality assurance, 2) can be transitioned to support activity, and 3) produce adequate design specifications which reflect supportability characteristics.

Software Project Management Implementation Methods

(CURRENT)
The software project management process utilizes implementation methods which enhance software supportability to the extent that implementation/coding/testing methodology, standards, and conventions are: 1) documented, followed, and validated through quality assurance, 2) can be transitioned to the support activity, and 3) produce supportable production products.

Software Project Management Organization Structure

(CURRENT)
The software project management process organization structure enhances software supportability to the extent that the physical structure, functional responsibilities, external interfaces and assigned personnel provide for continuity over the software life cycle phases, and have proper interfaces with organizations responsible for software support.

Software Project Management Planning

(CURRENT)
The software project management process utilizes planning which enhances software supportability to the extent that plans for the development, test, product transfer, operation and support exist, have been implemented, have been appropriately coordinated across activities, and satisfy contractual and/or regulation requirements.

Software Project Management Project Interfaces

(CURRENT)
The software project management possesses organization interfaces which enhance software supportability to the extent that external project organization relationships and responsibilities are: 1) defined, 2) provide a valuable functional role, and 3) contribute to systematic cost effective procurement, development, operation and support processes.

Software Project Management Test Strategies

(CURRENT)
The software project management process utilizes test strategies which enhance software supportability to the extent that the test plans, descriptions, procedures, and results have been:

1) documented, 2) can be transitioned to the support activity, and 3) provide for a consistent and systematic process for verifying and validating that software requirements have been satisfied.

Software Reliability

(CURRENT)
A quality of software which reflects the probability of failure free operation of a software component or system in a specified environment for a specified item.

Software Portability

(CURRENT)
A quality of software which reflects the effort required to transfer the software from one environment (hardware and system software) to another.

Software Support Concept

(CURRENT)
The estimated support personnel resources, level of dedication and expertise of the support personnel, and the duration of the block release cycle.

Software Support Facility (SSF)

(AFOTECP5)
The facility which houses and provides services for the support systems and personnel required to maintain the software for a specific ECS.

Software Support Personnel

(CURRENT)
See Support Personnel.

Software Support Resources

(CURRENT)
The totality of personnel, systems, physical facilities, and calendar time that are used/consumed during a software support release effort.

Software Supportability

(CURRENT)
A measure of the adequacy of personnel, resources, and procedures to facilitate:

(1) Modifying and installing software
(2) Establishing an operational software baseline
(3) Meeting user requirements.

Software Supportability Evaluation

(CURRENT)
An evaluation to derive a measure of how well a software system can be supported. (See Software Supportability.)

Software Supportability Evaluation Metrics

(CURRENT)
The closed-form questionnaire scores for each software supportability characteristic in a software supportability evaluation as well as the values computed by cumulating lower level scores.

Software Supportability Magnitude of Risk Consequence

(CURRENT)
The level of impact to a software user or supporter as a result of the risk level of a software supportability negative outcome.

Software Supportability Negative Outcome

(CURRENT)
Any outcome for which the software support resources are not adequate to accomplish required software support.

Software Supportability Risk

(CURRENT)
The probability at a given point during the software support phase that the software maintenance activity specified by a baseline software supportability profile cannot be accomplished with the available software support resources.

Estimated Software Supportability Risk: An estimate of the software supportability risk determined by the area under a normal distribution curve. The area is the part under the curve greater than the subject software's available person-months per change value as computed from the software support concept and baseline software change profile. The normal distribution curve is determined by using the estimated person months per change as the mean and the standard deviation from the derivation of the estimated person months per change regression equation.

Acceptable Software Supportability Risk: The estimated software supportability risk which is agreed upon by the user (using command) and supporter (supporting command) as a result of the baseline software supportability agreement.

Evaluated Software Supportability Risk: An approximation to the software supportability risk computed from the software supportability evaluation metrics. The computation is derived from a linear regression model using the software life cycle process, software product, support personnel, support systems, and support facility as the five regression equation factors.

Measured Software Supportability Risk: See Evaluated Software Supportability Risk.

Software System

(CURRENT)
A set of software (specifications, programs, and data) which constitutes a well-defined major function or group of functions.

Typical systems include avionics OFP, ground based communications, missile guidance, simulation, threat generator, ATE, and electronic warfare.

Software System Type

(CURRENT)
One of seven classifications of a software system's primary functional mission: ATD, ATE, C-E, EW, OFP, SIM, SUP.

ATD: Aircrew Training Device or Operational Flight Trainer for training and support of an operational system, usually in the form of a mockup simulator.

ATE: Automatic Test Equipment software to support the testing of hardware units under test (UUT), create and maintain the environment where the test software may be used, or prepare/analyze/maintain test software.

C-E: Communications-Electronics software for command and control, communications, surveillance and warning, air traffic control, intelligence, and other related functions.

EW: Electronic Warfare software that involves the use of electromagnetic energy and performs functions either separate or integral to a larger airborne or ground system.

OFP: Operational Flight Program software/firmware that is integral to an onboard aircraft computer system including navigation, flight control, fire control, weapon delivery, electronic engine control, and heads-up display.

SIM: Simulation Software not included as part of the ATD, including simulation models.

SUP: Support Software including application support software and system support software not included in any other category.

Specification Change Notice (SCN)

(CURRENT)
The SCN is used to distribute approved page changes to authorized users of baseline documents who, in turn, are responsible for posting the updates.

Source Code

(CURRENT)
The form of the program code in its source language.

Standards

AFOTECP3)
Procedures, rules, and conventions used for prescribing disciplined program design and implementation.

Support Concept

(CURRENT)
The software support concept usually specified as part of the CRISP and OS/CMP. Also includes that part of the support concept necessary to establish the acceptable risk from a baseline software change profile: standard release duration, number of support personnel, average skill level, percentage of personnel dedicated to releases, support facility, etc.

Support Facility

(CURRENT)
The physical facility resources that must be available for the software support resources to accomplish a specific task(s).

Support Personnel

(CURRENT)
A general term for personnel (military, DoD civilian, or DoD contractor) whose skills are necessary to directly support mission critical system software maintenance. Includes but is not limited to management, technical, non-technical support, and contractor personnel.

Support System

(AFOTECP5)
Any automated system used to change, test, or manage the configuration of ECS software and associated documentation. Includes but is not limited to Host Processor, Software Bench, Laboratory-Integrated Test Facility, Operational-Integrated Test Facility, and Configuration Management System.

Support System Facility

(AFOTECP5)
The facility resources that must be available for the software support resources to accomplish a specific task(s).

System Software

(AFOTECP5)
All of the software that is part of the software support facility computer system. It is never or seldom accessed directly by software support facility personnel; it controls the processing of application software. It includes the Operating System, Source Code Editor, Language Translator, Link Editor/Loader, Librarian/File Manager, Data Base Manager, and Automated Software Development Tool.

Test and Evaluation Master Plan (TEMP)

(AFR55-43)
An overall Test and Evaluation (T&E) plan designed to identify and integrate the effort and schedules of all T&E to be done in an acquisition program.

Threshold

(ROWE)
A discontinuous change of state of a parameter as its measure increases. One condition exists below the discontinuity, and a different one above it.

Time to Complete Maintenance Request (TC)

(CURRENT)
The calendar time from receipt of the maintenance request by the support control group until the request has been accepted as part of an operational system software configured release. (This does not mean the configuration is released or distributed, and this time does not include this additional delay, if any.)

Type

(CURRENT)
See Maintenance Type.

Uncertainty

(ROWE)
The absence of information; that which is unknown.

Urgent MA

(CURRENT)
See Maintenance Priority.

Verification/Validation (of computer programs)

(AFR800-14)
The process of determining that the computer program was developed in accordance with the stated specification and satisfactorily performs, in the mission environment, the function(s) for which it was designed.

END
DATE
FILMED
5-88
DTIC