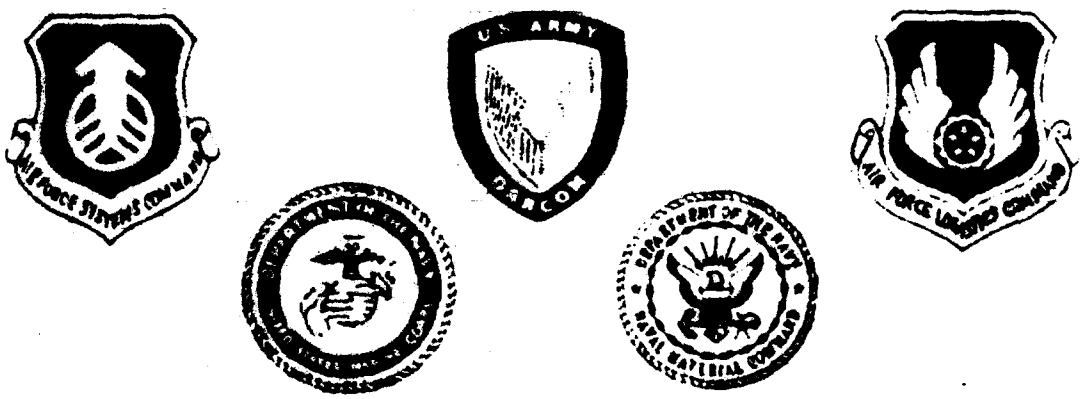
FINAL REPORT OF THE JOINT LOGISTICS COMMANDERS' WORKSHOP ON POST DEPLOYMENT SOFTWARE SUPPORT (PDSS) FOR MISSION-CRITICAL COMPUTER SOFTWARE

VOLUME I - EXECUTIVE SUMMARY

JUNE 1984



THIS IS NOT AN APPROVED JLC DOCUMENT

FOREWARD

The third biannual software workshop of the Joint Logistics Commanders' Joint Policy Coordinating Group on Computer Resources Management ~~is named "Orlando I"~~ and was held 31 October through 4 November 1983 at the Langford Hotel in Winter Park, Florida. The purpose of the workshop was to review Post Deployment Software Support (PDSS) activities for mission critical computer resources within the joint logistics commands and to make specific recommendations for uniform JLC policy relevant to PDSS life cycle support issues. Panels at the workshop addressed the issues of: 1) criteria for government/industry workforce mix, 2) independent verification and validation of computer software, 3) cost of present and future ownership of mission critical computer resources, 4) uniform software support environments, 5) policy manual for managing software change processing, and 6) configuration management requirements.

This volume presents a summary of objectives, findings, conclusions and recommendations of the six workshop panels. Volume II of this report presents the workshop proceedings which provide the details of the workshop organization, summaries of guest speaker presentations, and the complete panel reports for all six of the workshop study groups.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ORLANDO I WORKSHOP

1. INTRODUCTION

1.1 BACKGROUND

Appreciating the growing importance of digital computer resources including computer software in the development and support of weapon systems, the Joint Logistics Commanders instituted in 1977 a Joint Policy Coordinating Group on Computer Resource Management (JPCG-CRM). The mission of this body is to coordinate and insure consistency in the preparation of new and revised regulations and standards, to provide recommendations on critical resource areas and to provide a focal point for coordinating standardization programs. To address software related issues, in 1978 the Computer Software Management (CSM) subgroup was formed subordinate to the JPCG-CRM. This organizational relationship is shown in Figure 1. The specific mission of the CSM subgroup is "to review policies, procedures, regulations and standards relating to computer software and to forward acquisition and management, including software development, quality, testing and post-deployment support.

To accomplish their mission the CRM has organized three very significant, joint government/industry workshops attended by experienced computer resource practitioners. The first workshop called "Monterey I" was held in 1979 at the Naval Post Graduate School at Monterey, California. Monterey I was concerned with software development and acquisition issues such as policy, software development standards, software documentation standards, software quality assurance standards and acceptance criteria. Two years later at "Monterey II" these issues were reviewed once more along with new areas of concern relating to computer resource configuration item selection, standardization and accreditation of computer architectures, software cost estimating, and software reuseability. The products of these two conferences are stabilizing through further government/industry interchange and are beginning to be used in defense system acquisitions. These two important workshops led to draft Department of Defense software development standards, a tri-service software and standard data item descriptions (DIDs) that are expected to be formally implemented in late 1984. Furthermore, military standards on reviews and audits (MIL-STD-1521), engineering specifications (MIL-STD-490), and configuration management (MIL-STD-483) were revised to include improved and compatible software engineering and management requirements. A draft software quality assurance management standard (MIL-STD-SQAM) was developed and reviewed in conjunction with industry. Government and industry have agreed to an approach to resolving MIL-STD-SQAM issues. A revised software quality standard and policy are expectd to be implemented in late 1985.

The third biannual workshop, labelled "Orlando I," was held in late 1983 and is the focus of this report. Whereas Monterey I and II dealt with software development and acquisition, Orlando I focused on the support of mission-critical computer resources after its initial development and deployment. Figure 2 from the Monterey conferences presents an idealization of the computer software development cycle as it often relates to the system acquisition phase. These activities, however, have unique problems over those seen during acquisition. The purpose of Orlando I was to identify and to define clearly some of these problems and their solutions so that an action plan to address them may be prepared.

JOINT LOGISTICS COMMANDERS' [JLC] ORGANIZATION

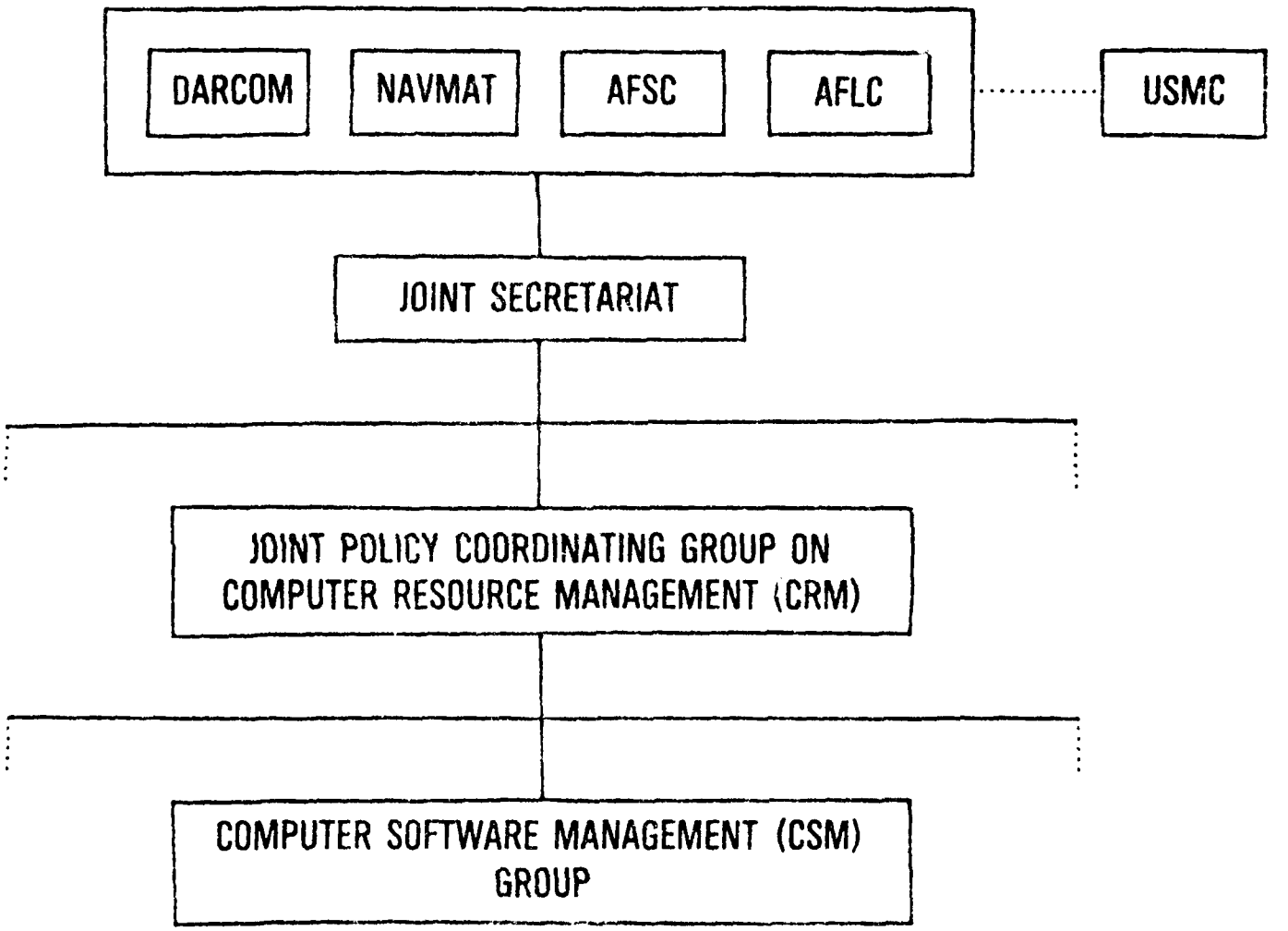


Figure 1

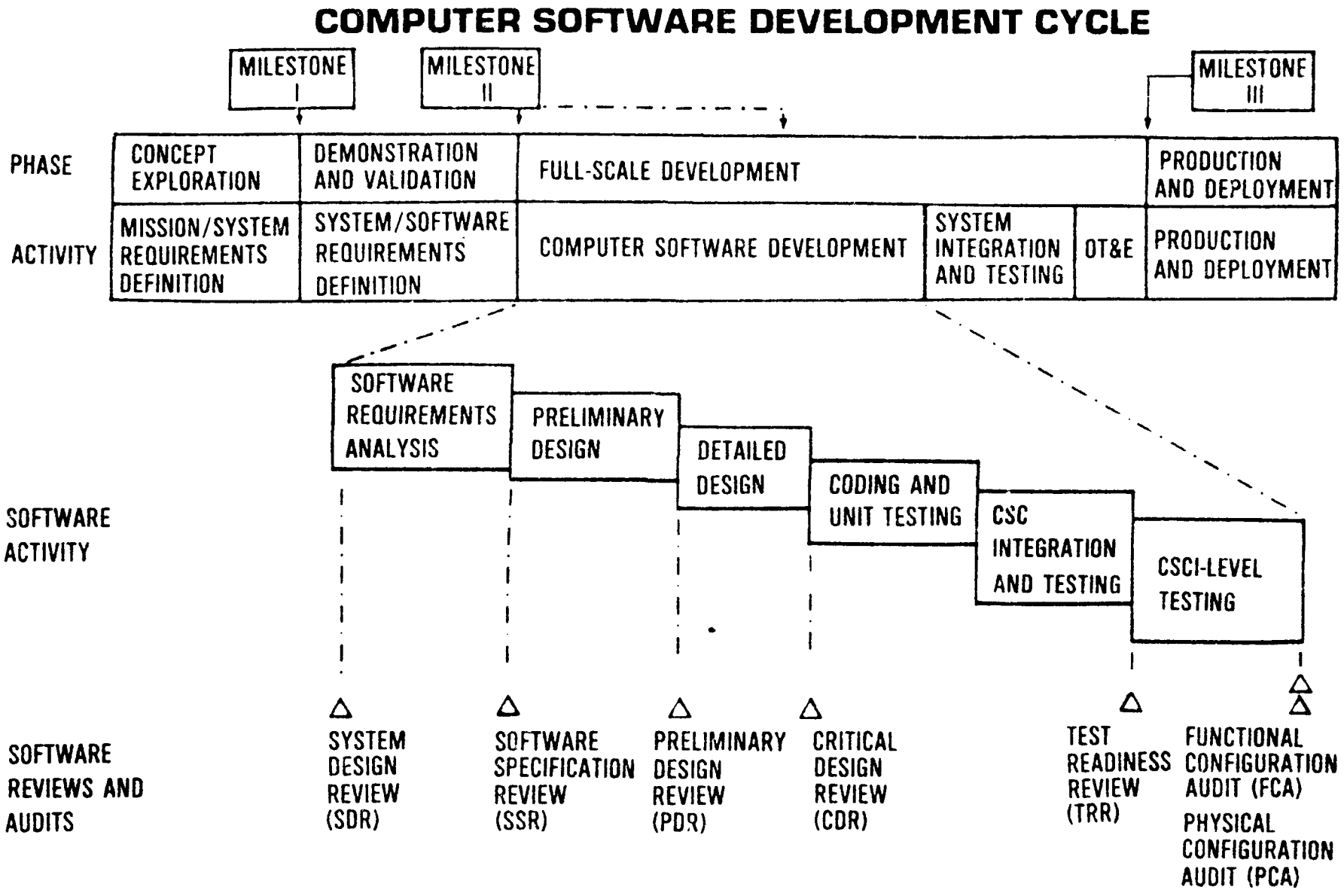

COMPUTER SOFTWARE DEVELOPMENT CYCLE
MILESTONE I
MILESTONE II
MILESTONE III
PHASE
CONCEPT EXPLORATION
DEMONSTRATION AND VALIDATION
FULL-SCALE DEVELOPMENT
PRODUCTION AND DEPLOYMENT
ACTIVITY
MISSION/SYSTEM REQUIREMENTS DEFINITION
SYSTEM/SOFTWARE REQUIREMENTS DEFINITION
COMPUTER SOFTWARE DEVELOPMENT
SYSTEM INTEGRATION AND TESTING
OT&E
PRODUCTION AND DEPLOYMENT
SOFTWARE ACTIVITY
SOFTWARE REQUIREMENTS ANALYSIS
PRELIMINARY DESIGN
DETAILED DESIGN
CODING AND UNIT TESTING
CSC INTEGRATION AND TESTING
CSCI-LEVEL TESTING
SOFTWARE REVIEWS AND AUDITS
SYSTEM DESIGN REVIEW (SDR)
SOFTWARE SPECIFICATION REVIEW (SSR)
PRELIMINARY DESIGN REVIEW (PDR)
CRITICAL DESIGN REVIEW (CDR)
TEST READINESS REVIEW (TRR)
FUNCTIONAL CONFIGURATION AUDIT (FCA)
PHYSICAL CONFIGURATION AUDIT (PCA)

Figure 2

1.2 ORLANDO I WORKSHOP OBJECTIVES

The primary objective of the Orlando I workshop was to identify and record interservice policies, problems and approaches to six specific areas of "post deployment software support." The workshop members generated panel reports in each of the six areas recording the assessments of the panel participants, their recommendations and their guidance. These reports, presented in their entirety in Volume II of this document, are to serve as the basis for a JPCG-CRM/CSM action plan that describes the required actions, resource, schedule, and responsible organizations for implementing the JLC approved recommendations.

The six specific areas assigned to the panels comprised:

1) government/industry workforce mix
2) independent verification and validation
3) cost of ownership
4) software support environment
5) the software change process
6) software configuration management

A specific, primary panel objective in each of these areas is summarized in Table 1. A summary of the results of the panel investigations in accord with these objectives is presented in Section 2, "Findings and Recommendations," of this report.

1.3 WORKSHOP ORGANIZATION

The Orlando I Joint Logistics Commanders' workshop on Post-Deployment Software Support was held from 31 October through 4 November, 1983, at the Langford Hotel, Winter Park, Florida. The management level organization of this third software workshop is shown in Figure 3, while the administrative committees are shown in Table 2.

Each of the six panels was co-chaired by government and industry chairpersons, who worked together to execute the pre-established agendas for the panels. These chairpersons are identified in Table 3. At the completion of each day's sessions, minutes of deliberations were prepared and reviewed. Outlines and preliminary drafts of panel reports were available at the end of the workshop. Each participant also completed an evaluation form regarding the workshop utility at the conclusion of the workshop.

During the several months following the workshop, the draft panel reports were reviewed by all panel members, were revised based on panel member comments, and were reviewed once more. The final panel reports are found in Volume II of this report. A summary of findings and recommendations of the panels is found in the following section. Furthermore, a strawman charter for PDSS activity is presented in Appendix A of this executive summary.

ORLANDO I

Mission-Critical Post Deployment Software Support (PDSS) Workshop

PANEL OBJECTIVES

Panel A - INDUSTRY/GOVERNMENT WORKFORCE MIX
Develop policy recommendations for cost-effective staffing of software support agencies using appropriate mixes of government and industry personnel.

Panel B - INDEPENDENT VERIFICATION AND VALIDATION (IV&V)
Determine when and how much IV&V should be used in software development and during Post Deployment Software Support (PDSS).

Panel C - COST OF OWNERSHIP
Clarify the basis of large projected costs of future software development and support while identifying approaches to reducing software cost.

Panel D - SOFTWARE SUPPORT ENVIRONMENT
Discuss the requirements for establishing an effective, generic post deployment software support environment establishing feasibility, advantages and disadvantages.

Panel E - THE SOFTWARE CHANGE PROCESS
Develop the framework for a joint services PDSS "Change Policy Manual."

Panel F - CONFIGURATION MANAGEMENT
Determine a common definition and scope of "software configuration management" which is suitable to be promulgated by the JLC.

Table 1

THIRD SOFTWARE WORKSHOP MANAGEMENT

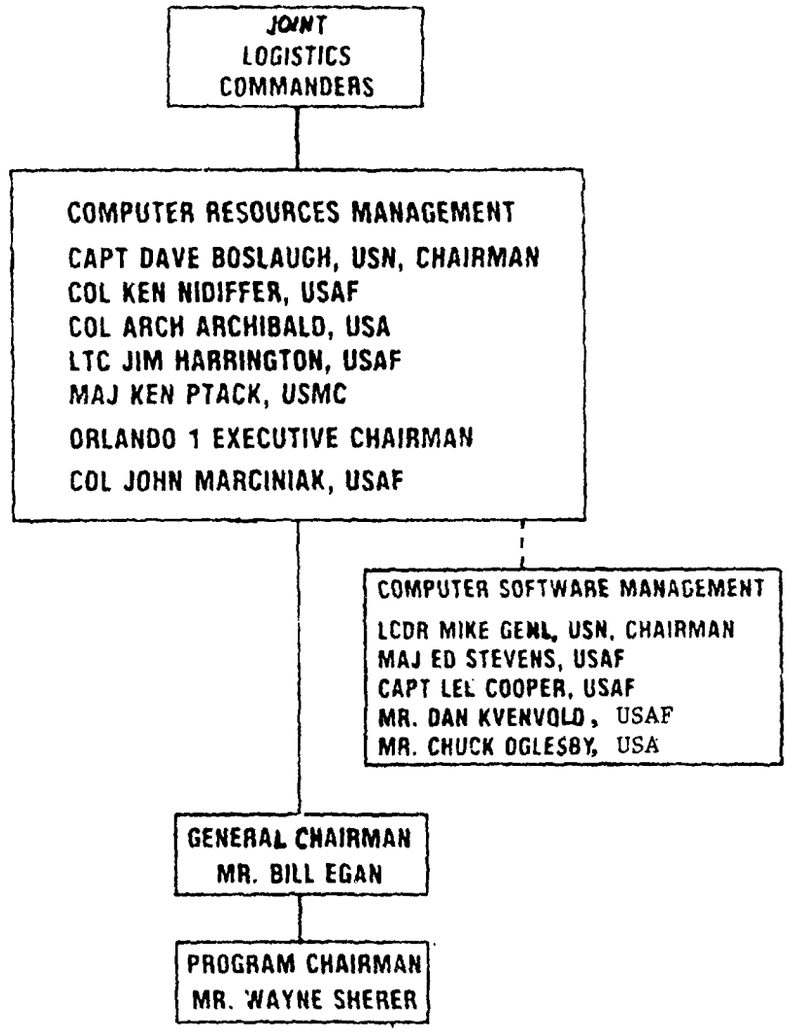


Figure 3

ADMINISTRATIVE ORGANIZATION

Executive Chairman:
Colonel John Marciniak, USAF

Executive Committee:
Capt. Dave Boslaugh, USN
Col. Ken Nidiffer, USAF
Lt. Col. James Harrington, USAF
Col. H. R. Archibald, US Army
Maj. K. R. Ptack, USMC

General Chairman:
Mr. Bill Egan, Naval Air Systems Command

Program Chairman:
Mr. Wayne Sherer, U.S. Army Armament Munitions & Chemical Command

Facilities Chairman:
Capt. Tom Smith, US Marine Corps

Publications Chairman:
Maj. Ed Stevens, HQ AFSC/ALR
Capt. Lee Cooper, HQ AFSC/ALR

Special Arrangements:
Mr. Mert Batchelder, HQDARCOM

Protocol Officer:
Lt. Sunny Riley, HQAFLC/MMEC

Administration/Business Manager:
Ms. Roxy McCarter, HQNAVMAT

NTEC Liaison:
Mr. Frank Jamison, Naval Training Equipment Center

Workshop Manager:
Ms. Michele Foley, P/M Group

Planning Support:
Ms. Dreama Fumia, Veda, Inc.

Treasurer:
Mr. Daniel Kvenvold

Table 2

PANEL CO-CHAIRPERSONS

Panel A - Government/Industry Workforce Mix

Lt Col Frank J. Sisti
HQ DA (DAMO-C4L) Pentagon
Washington, DC 20380
(202) 697-4539
A/V 227-4539

Mr. R. Dean Hartwick
Logicon, Inc.
255 West 5th St.
San Pedro, CA 90731
(213) 831-0611

Panel B - Independent Verification and Validation (IV&V)

Cdr D. (Dave) Southworth
HQ, Naval Material Command (MAT 08Y)
Washington, DC 20360
(202) 692-3966

Mr. John W. Sapp
Software A&E, Inc.
1401 Wilson Blvd., Suite 1220
Arlington, VA 22209
(703) 276-7910

Panel C - Cost of Ownership

Lt Col James Riley
HQ AFSC/DLA
Andrews AFB
Washington, DC 20334
(301) 981-2482

Mr. G. (Gene) Sievert
Teledyne-Brown Engineering
300 Sparkman Dr.
Huntsville, AL 35807
(205) 532-1500

Panel D - Software Support Environment

Mr. Jim Hess
HQ DARCOM/DRCDE-SB
5001 Eisenhower Avenue
Alexandria, VA 22333
(202) 274-9318
A/V 284-9318

Mr. Jerry Raveling
Sperry Corporation
Computer Systems, M.S. U1E13
P.O. Box 43525
St. Paul, MN 55164
(612) 456-3545

Panel E - The Software Change Process

Mr. Joe Black
WR-ALC/MMRR
Robins AFB, GA 31098
(912) 926-5948

Mr. Jack Cooper
CACI, Inc.
Federal Penthouse
1700 N. Moore St.
Arlington, VA 22209
(703) 276-2826

Panel F - Configuration Management

Mr. C. (Cal) Showalter
Naval Air Systems Command
(AIR-543C)
Room 620, JP-2
Washington, DC 20361
(202) 746-0650

Ms. Antonia D. Schuman (Toni)
TRW Systems Group
1 Space Park, Bldg. 134
Room 6079
Redondo Beach, CA 90278
(213) 217-4079

Table 3

2. FINDINGS, CONCLUSIONS, AND RECOMMENDATIONS

Mission critical computer systems are defined by P.L. 97-86 (the Warner Amendment) and the associated OSD implementation guidance as those computer resources or services involved in the function, operation, use or R&D of intelligence systems, cryptologic systems related to National security, command and control of military forces, weapon systems or other systems critical to the direct fulfillment of military or intelligence missions.

As the government identifies the need for and procures increasing numbers of mission-critical computer systems, "Post-Deployment Software Support" must have a high priority in planning. A precise definition of PDSS is necessary to form the basis of consistent policy and planning. The following definition was developed at Orlando I:

> "Post-Deployment Software Support (PDSS) is the sum of all activities required to ensure that, during the production/deployment phase of a mission-critical computer system's life, the implemented and fielded software/system continues to support its original operational mission and subsequent mission modifications and product improvement efforts."

Some panels preferred to use the term "Post-Development Software Support" for the symbols PDSS. The term "software maintenance," however, was considered to be inadequate to convey the true nature of software support. Maintenance consists primarily of the activities and methods of restoring something that is broken to its original "unbroken" form. "Software support" is directed both at software redesign to correct software errors and to enhance current features or to add totally new functions. "Software maintenance" simply does not convey either of these functions properly and therefore should not be applied to PDSS. Specifically the term "Post-Deployment Software Support" was chosen over the term "Post-Development Software Support" so that the focus of the workshop was primarily on the phase of support when the mission-critical computer system is seeming operational rather than focus on the activity immediately after the development but before military software support is fully active.

The findings, conclusions and recommendations on government/industry workforce mix, independent verification and validation, cost of ownership, software support environments, the software change process, and configuration management as they relate to PDSS are summarized in the following six sections.

2.1 GOVERNMENT INDUSTRY WORKFORCE MIX

The objective of the panel addressing appropriate PDSS Government/Industry Workforce Mixes was to develop policy recommendations for cost-effective staffing of software support agencies. In pursuing this objective, the panel identified problems in PDSS planning and funding as well as a lack of accepted triservice criteria for arriving at an appropriate workforce mix.

2.1.1 Findings and Conclusions

2.1.1.1 Planning

Regarding planning, the entire weapon system life cycle must be

accounted for throughout all phases of the developmental process. Each service approaches this planning requirement in a similar fashion. The plan to accomplish the mission of developing/producing/fielding/maintaining a system includes a plan on how to acquire and manage computer resources. Each service has a regulation that describes this plan.

The Army's Computer Resources Management Plan (CRMP), the Navy's Software Life Cycle Management Plan (SLCMP) and the Air Force's Computer Resources Integrated Support Plan (CRISP) specify the elements for PDSS of the system, reflecting schedules, resource allocations, organizational interactions, and activity responsibilities associated with the project's life cycle. The Monterey workshops recommended a Computer Resource Life Cycle Management Plan (CRLCMP) which is the JLC nomenclature for a generic plan acceptable to all services. The current JLC software development standardization project includes policy on and a DID for the CRLCMP.

In spite of having specific plans, it is apparent that all services have problems in using their existing life cycle management plans. Specific problems include the following:

1) The plans do not cover all the necessary information that they should.

2) The plans are often not required early enough in a system's life cycle so as to impact the resourcing of the system.

3) The regulations which require computer resources documentation do not provide for sufficient discipline in the management process to ensure that plans are submitted as required.

4) Users of the potential plan do not understand that an early (in the system life cycle) management document must be a true living document, and hence they do not plan for updating it.

To properly plan a workforce mix that is attainable and achieves all goals for a given support system, the computer resources planning document must be on hand and must be used. Such a document, even as it evolves from one life cycle phase to another, must exist and must have widespread distribution among impacted activities/agencies. Coordinated and approved modifications to that document must, therefore, receive equal visibility and distribution. There seems to be no need to modify the definitions of the CRLCMP, CRMP, SLMPC, or CRISP. However, the panel unanimously concluded that these documents, if developed, are thereafter either ignored or never updated as a program progresses through the acquisition cycle. The result is that the PDSS tends never to be properly considered or planned at the time of system deployment. Accordingly, the panel concluded that the JLC should establish procedures to provide for the proper use and update of the computer resource plans by all services.

2.1.1.2 Funding

A problem of how PDSS is funded occurs commonly across the services. PDSS funding is almost always fragmented, making it difficult to manage properly. For example, system acquisition and PDSS in the Air Force are budgeted and funded

through separate channels and processes (AFSC and AFLC). Even after program transfer, hardware and software are budgeted, funded, and prioritized by separate processes. This creates confusion as to the proper acquisition process, clouds actual cost tracking, and requires careful coordination of one-year software money with three-year hardware money for the system modification. The Navy has similar problems in that a large portion of development and functional enhancement to a weapon system is done using Operational Maintenance (OMN) funds and Advance Procurement (APN) funds. If some funds are marked for multiple years and others must be obligated or outlaid within one year, contracting for PDSS tasks must be partitioned to accommodate this funding cycle. Task coordination and schedule interfaces become difficult, and schedule delay or cost growth results. The proper allocation of dollars to functional tasks would improve the contracting posture and schedules of the PDSS function.

Streamlined policies and procedures are needed for budgeting and funding the development, acquisition, and support of mission-critical computer resources. These should provide common budgeting and funding procedures among the services for presentation to the President and Congress, identification of appropriations, budget programs, program elements and specific fund codes to weapon systems, a single prioritization process, and simplification of procedures.

2.1.1.3 Workforce Mix

The various combinations of government/industry workforce mixes can be summarized in three "most likely" PDSS organizations: organic support, developer support, and independent support contractor (ISC) support. For organic support, PDSS is assigned to an organic activity within one of the military departments. In some cases, the organic support activity reports to a system project manager and employs an optimum mixture of military, civil service, developer contract support, and/or support services contractors to accomplish the PDSS mission under the direction of the organic support activity. For developer (only) support, PDSS is contracted to the original developer for total PDSS support with direction provided by the designated project/functional manager. In the final alternative, PDSS is contracted to an independent support contractor for total PDSS support with direction provided by the designated project/functional manager.

The attributes that drive the selection of PDSS personnel are as follows:

1) User Oriented
2) Logistics Oriented
3) Technically Oriented
4) Personnel and Resources Oriented
5) Administrative/Politically Oriented

With a few exceptions, the panel determined that there is no a priori attribute that drives the assignment of a workforce. A few attributes do tend to direct the use of specific personnel. For example, those attributes reflecting control (e.g., configuration management) would appear always to require either military or government civilian participation. The less mature and more complex the system, the more participation is required by the original developer. Industrial participation by other than the original developer is driven only by either the need for additional staff (where military/civil service cannot be supplied) or by the need for lower cost (either original developer or civil

service).

From these observations, the panel concluded the following:

1) A military presence at some low-level of effort is required to provide continuity and user influence and to govern embedded doctrine.

2) Government civilian personnel are required to provide technical capability as necessary to maintain government control and to provide an enduring corporate memory.

3) The original developer's participation is always required at fairly high levels on complex and immature software, and then that participation dwindles as the software matures.

4) Support contractors can provide additional technical services not available through the government, or can lower the cost of PDSS.

The panel's determination of how government/industry personnel should be allocated does not markedly differ from the way the allocations are now generally made by the services. Certain minimum requirements exist for on the order of 20% of the PDSS staff to be government and a number of staff to be supplied by the original developer (this number decreases as the software matures). The majority of the PDSS staff (approximately 80%) are then assigned from either civil service or industry based upon the particular needs/availability/funding or political outlook of the manager.

2.1.2 Recommendations

The panel formulated a recommendation to the JLC regarding the PDSS planning and a recommendation on the PDSS funding. These are as follows:

1) The JLC should establish procedures that ensure that the PDSS provisions in the new CRLCMP or its existing counterparts (US Army-CRMP, US Navy-SLCMP, and US Air Force-CRISP) are complied with at the outset of all software acquisitions, and ensure that the PDSS provisions are upgraded throughout program acquisition. This plan should be included at all service and DOD program reviews, including system acquisition review councils (e.g., DSARCs).

2) The JLC should streamline policies and procedures for budgeting and funding the development, acquisition, and support of mission-critical computer resources.

No recommendation by the panel to the JLC was advocated regarding the government/industry workforce mix; however, considerably more discussion on the technical aspects of the appropriate mix is found in the complete panel report in Volume II.

2.2 INDEPENDENT VERIFICATION AND VALIDATION

Independent Verification and Validation (IV&V) is defined as verification and validation of computer software performed by an organization that is managerially and financially independent from the developing organization. Panel B was chartered to study when and how much IV&V should be used in software development and in Post Deployment Software Support (PDSS). This is an extremely complex issue because of the overlapping roles of test, quality assurance and systems engineering.

2.2.1 Findings and Conclusions

The following provides a summary of the major findings of Panel B concerning IV&V.

1) IV&V is beneficial based on a cost/benefit analysis. These benefits are quantifiable and should be considered in all programs. The panel identified generic costs and benefits and determined those that could be quantitatively modeled. Existing methodologies were used to develop an expression of the cost/benefit ratio (See Appendix H of Panel B report in Volume II). Additionally, costs and benefits, which could be stated in qualitative terms, were identified and reinforced through "case study" experience of the panel members. Further refinements of the estimation model are required (see recommendations).

2) IV&V can and should be used in all phases of the software development cycle. These IV&V activities are the same in PDSS as in the other phases of the life cycle. The level of IV&V in PDSS should be determined using the same criteria as in the other phases.

3) It is beneficial to begin the IV&V effort as early as possible in the development cycle. As can be shown in both the cost/benefit model and the qualitative factor analysis, the earlier an error is discovered, the less it costs to fix. Another benefit of IV&V in the early stages of development is to catch overlooked errors early. This has merit, especially in the design stages where errors are identified well before they are "locked" in actual code.

4) The level of effort for IV&V can be measured on discrete levels based on specific criteria and degrees of risk. Models can be developed which will give program managers specific guidance on how much IV&V to use. These levels of effort are identified as Bare Bones, Low, Moderate and Full Blown (see Appendix J of Panel B report in Volume II). The extent to which IV&V should be implemented is based upon specific criteria (e.g., complexity, mission essentiality, safety, etc.) and the degree of risk (high, medium, low) that each criterion places on the project. The risk factor is a measurement of the degree of impact each criterion has on the overall software development. Further refinement of the model is required to assess the levels of risk, to weight the criteria and to map the results so as to determine the proper levels of IV&V to use (see recommendations).

5) IV&V must be adequately financed to support the level of effort decided upon. The program manager could use the cost/benefit analysis methodology to justify the need for IV&V and then use the guidance models to ascertain the amount and types of IV&V effort. He could then strike a balance between program funding and the recommended IV&V effort to arrive at a decision on the amount and type of IV&V to be done with available resources.

6) IV&V can be done "in-house" or by a separate contractor as long as the IV&V agent is independent of the developer. The attribute of independence requires that IV&V be performed by an organization that is managerially and financially independent of the developing organization. Such a separation is necessary to provide a basis for an objective V&V activity and to provide accountability to those responsible for acquisition of the software.

7) Experience in IV&V and possession of and experience with the proper tools is the best predictor of an organization's future success in an IV&V environment. Further, it was felt by the panel that the IV&V staff's skill and qualifications are a more critical ingredient than the IV&V tools used. Specialization in IV&V by an organization should therefore be a prime requisite when selecting an IV&V agent.

8) The PDSS activity should be involved in the IV&V effort as early in the development cycle as possible. In this way personnel responsible for post deployment software support can be trained on the system well before it is turned over to the software support activity. In fact, the PDSS activity should be the preferred activity to conduct IV&V of the system while it is being developed because of that training effect and because the ongoing nature of a PDSS activity can give it the chance to become an IV&V specialist.

2.2.2 Recommendations

Based on the above findings, the following seven recommendations are submitted to the JLC.

1) The draft Joint Policy, Software Quality Program, dated 1 October 1982, contains the following definition of IV&V:

> Independent Verification and Validation. The verification and validation of computer software performed by an organization that is managerially and financially independent from the developing organization.
>
> Validation. The evaluation, integration, and test activities carried out at the system level to ensure that the finally developed system satisfies the using command's mission requirements set down as performance and design criteria in the system specification.
>
> Verification. The iterative process of determining whether the product of each step of the computer software development process fulfills all requirements levied by the previous step.

The panel recommends that the above definition of "Validation" be slightly modified by changing the wording to emphasize software and the software support environment. The following modified wording is recommended:

> Validation. The evaluation, integration, and test activities carried out at the system level to ensure that the finally developed Computer Software Configuration Items (CSCIs) satisfies the user's and supporter's requirements set down as performance

and design criteria in the system and software requirements specifications.

2) JLC policy should state that program managers should determine the extent of IV&V effort to be used in their program as part of an overall program trade-off analysis. This policy should be incorporated as part of a DOD Directive or Instruction and made part of the acquisition process as a check-off item for review boards and acquisition review councils. The purpose of such a policy would be to focus high level concern on the IV&V activity, to ensure that proper funding for IV&V activities is considered at an appropriate stage of a system's life cycle, and to promote a consistent application of IV&V across the DOD software acquisition spectrum.

3) A program manager's (PM) guidebook should be developed to help the program manager to accomplish the following:

- o Complete a cost/benefit analysis
- o Determine the level of IV&V to be done
- o Determine what IV&V efforts should be accomplished during various phases of the life cycle

The guidebook should instruct the program managers and their staffs in the methodology of the analyses so that IV&V requirements are credible and consistent.

4) The JLC approve further selective data collection for the cost/benefit model improvement and calibration activities. This will provide the PM with a more precise resource prediction capability. The model needs to be refined and validated in a controlled environment to improve the precision of the estimate. With a well accepted and proven prediction model, PMs will possess more credibility in their resource request and will be able to consider seriously the use of IV&V in their software development.

5) The JLC approve further selective data collection for the refinement of the criterion model for selection of the levels of effort for IV&V. Further research is necessary in the areas of the criteria employed, weighting schemes for levels of risk and for the criteria, as well as incorporating these schemes into a selection model. With this model at the PMs disposal, a precise analysis of the amount of IV&V to be used can be done. This will result in better resource management and increased overall program efficiency.

6) The JLC endorse separate IV&V responsibilities within the Acquisition Commands. The emphasis here is on a true independence of the IV&V activity from those that develop the software. This can be accomplished in many different ways, as is described in the body of the report. More emphasis should be placed on IV&V as a separate requirement and as a separate activity. Even though the tools, processes and activities of IV&V, and software quality assurance and measurement overlap in function, the purpose behind each is different. IV&V should be emphasized as an independent process. Further JLC effort should be expended to define more clearly the purpose and activities of IV&V and to define its role in relation to Software Quality Assurance and measurement.

7) The JLC approve the need for further study as specified in each

subpanel report and include these in follow-on workshops. IV&V is a critical and very comple area of concern in the software development cycle. As such, there are still many areas that require further study and discussion and which deserve JLC visibility. Included in these, in addition to those described above, are the following:

- How does the "reuseability" of software issue impact on the need for IV&V?
- What IV&V tools are required for software engineering environments? Are there Ada unique requirements? What criteria should be used in the selection of these tools?
- How does the expansion of the use of firmware impact on the need for IV&V and its methodology.
- Do distributed systems differ in their need for IV&V from centralized applications?
- What contractual mechanisms can be applied to ensure adequate IV&V?
- What IV&V procedures are appropriate for determining when software should be redesigned rather than continuing to modify it?
- Are there any additional formal reviews that need to be established as part of PDSS?
- Are there unique IV&V requirements for security functions?
- What documentation is required for the IV&V effort?

2.3 COST OF OWNERSHIP

The Cost of Ownership Panel was chartered with achieving an understanding of the true life cycle cost of ownership of DOD software; with identifying actions which can be taken under JLC auspices to make it possible to identify, track and control those costs; to investigate the utility and feasibility of a common DOD PDSS Center charter and draft such a charter if appropriate; and to recommend to the JLC actions which, if taken by the services, might significantly reduce software ownership costs. The findings, conclusions, and recommendations of this panel are presented below.

2.3.1 Findings and Conclusions

The point of departure for both panel and subpanel discussions was a series of four briefings on software ownership cost. Mr. Pat Mellin presented a briefing which was prepared in 1980 as a result of a study sponsored by the Electronics Industry Association (EIA) on the cost of DOD digital data processing. The conclusion of most interest to the panel was that the total annual cost of ownership of DOD embedded computer software would rise to approximately $32 billion by 1990. This briefing was followed by three presentations on software costs within the services. The estimates based on Army and Navy data, presented

by Gene Sievert and Bill Smith respectively, were arrived at by parametric analysis and were generally consistent with the EIA forecast. The Air Force presentation by Jerry Schmidt, on the other hand, reflected actual POM submissions based on projections of systems to be supported by both AFLC and the using commands (SAC, TAC, etc.). When these figures were adjusted for inflation and extrapolated to account for AFSC development costs, the Air Force number was significantly lower than the EIA projection would indicate.

This variance among estimates triggered a lively discussion which pervaded all further deliberations at the panel and subpanel levels and in fact spilled over into casual conversation. This intense concentration on the cost prediction issue ultimately enabled the panel to reach consensus in addressing the two panel-level goals:

1) determine the credibility of DOD 1990 predicted embedded computer costs

2) determine the cost of maintaining post-development embedded software systems

The panel finally agreed unanimously that while the growth rate in embedded software in the short term will be as high as implied by the EIA study, that growth rate will not be sustained through the 1980's. Thus, the $32 billion estimate for 1990 is probably high. We also agreed that we do not currently have the data to offer an alternative figure to the $32B, but that the PDSS portion of that cost would probably be between $5B and $7B in 1990.

All major subpanel goals were achieved. One subpanel compared the current service approaches to many detailed PDSS activities, and concluded that despite some different views relative to management and funding procedures there is enough internal similarity to make a common PDSS center charter useful. A second subpanel drafted an excellent strawman for a common PDSS center charter. The third subpanel agreed upon and documented the physical facilities required by a generic PDSS center, including requirements to address security considerations.

2.3.2 Recommendations

The Cost of Ownership Panel recommends that the JLC sponsor the following:

1) A triservice effort to identify the real cost of software for a near-term future baseline fiscal year.

2) Changes in procurement regulations to force the use of work breakdown structures which clearly separate all software and system engineering tasks from hardware related tasks.

3) Changes in contracting methodologies and procurement regulations to require contractors to report costs against these WBS's.

4) Changes in DOD accounting practices to make it possible to ascertain direct DOD software costs.

5) A solution to the problem of multiple appropriations (R&D vs

O&M) and funding lines to support software evolution after transition. A new funding line to provide for evolutionary support after transition should be established. (A minority of two panel members agreed that a new funding line for evolutionary software development would be the ideal solution, but felt that the difficulties in establishing a new appropriation could well outweigh the benefits. The recommendation of this minority was that the JLC sponsor a tradeoff study to balance the cost of justifying and establishing a new appropriation against its potential benefits.)

6) DOD should direct its efforts to optimizing the expenditure of DOD resources even if it means rapidly expanding software growth rates.

7) JLC should review policies governing acquisition requirements for adequate coverage of software life cycle support requirements and tighten procedures for promoting adherence to these policies.

8) OSD should regain control and seriously examine the issue of Ada environments to determine the most cost effective method of continuing with the original intent of cost reduction through an Ada standard implementation.

9) JLC should identify specific program development areas which could benefit from application of available or near mature automation tools and begin to utilize these in specific applications hand in hand with cost data tracking
and management.

10) JLC should institute a program to develop procedures, organization elements, policies and support tools necessary for reuseability, and identify program areas of high software reuseability potential to participate in such an initiative.

11) JLC should direct the adoption of the strawman charter for a common PDSS (Attachment A)

2.4 SOFTWARE SUPPORT ENVIRONMENT

Over the past decade there has been a dramatic increase in the number of planned and deployed Mission-Critical Computer Systems (MCCS). A MCCS is a system which is of significant importance and which is integral to the effectiveness of today's military combat and support systems. MCCS's implement or aid in the implementation of system and subsystem performance characteristics, and serve to integrate the various system elements into highly responsive and effective systems. MCCS's, through their programmability features, provide military systems with improved flexibility to respond to changing operational requirements.

With the continued improvement in the cost/performance ratio for computer hardware, and improvements in computer software capabilities, the military services are able to develop and deploy more-and-more complex systems. At the same time, this dramatic expansion in the use of MCCS is creating new and continually expanding logistic support requirements. All of the Services are confronted with the problem of supporting a rapidly expanding number of unique computer based systems. Each unique MCCS, brings with it its own Instruction Set Architecture (ISA), hardware spare parts requirements, and related support and application software. The logistics support problem for MCCS has been further

exacerbated in recent years through the introduction of microprocessor based embedded subsystems/systems.

MCCS software serves to modify, enhance, and integrate the processing system into a functional system. The MCCS software controls the capability of the system. Today's military, in most instances, can not perform their mission without full reliance upon the MCCS software which is inherent to their operational systems.

To effectively and efficiently modify MCCS software and, in general, provide engineering support for the MCCS, requires specialized facilities, skills, and equipment. After the acquisition of the operational (or test, training, etc.) system has been completed and the system has been deployed to its operational environment, the military services commands/organizations assume responsibility deployment engineering and support.

The principal difference in post-deployment Software Support Environments (SSE's) is related to the basic maintenance concept established for a system, and its major subsystems. That is, will support be centralized or decentralized and what level of system (or subsystem) support will be provided?

The panel's basic objectives were to define the requirements for a generic PDSS software support environment, and to assess the commonality of requirements with DOD-sponsored, development-oriented software environments.

The panel was assigned to discuss selected aspects of a generic PDSS environment. These aspects were addressed to the panel in the form of a series of questions which dealt with:

1) Requirements for defining a core SSE generic equipment/software suite.

2) Management support system requirements to include criteria for GFE/CFE, security, and PDSS versus development environments.

3) Major contractual considerations which must be addressed in the system acquisition and post-development phases of the life cycle.

4) Whether the type of software to be supported by the PDSS facility places unique requirements on the SSE.

2.4.1 Findings and Conclusions

1) The commonality factors and economics of scale of a single unified SSE for a referenced system outweigh the advantages of having several SSEs support different phases of the software life cycle.

2) Other than the distribution management functions required for PDSS, there are no significant differences between the generic functions required for a development mode SSE and a post-deployment mode SSE. Any SSE function which provides effective support in one mode also provides effective support in the other mode.

3) The development SSE and the post-deployment SSE are almost identical. Primarily, the post-deployment SSE must grow to support distribution management and many defense system functions and capabilities not identified during development.

4) From a DOD ownership standpoint, it is extremely important to establish interface definitions between components of the Software Engineering Environment (SEE) to promote commonality, interoperability, and evolution among SSE's.

2.4.2 Recommendations

1) The Computer Resources Life Cycle Management Plan (CRLCMP) should identify the directions of evolution which the SSE will need to support, identify the organization responsible for supporting the post-development evolution of the SSE, and provide a clear transition plan between the development SSE and the post-development SSE.

2) The current STARS SEE effort should develop an initial definition of the interface between components of its SSE and support further R&D toward a more complete definition.

3) The JLC should sponsor a study of long-term PDSS security requirements with emphasis on Ada run-time environments, and other PDSS unique requirements.

4) The acquisition agency must insure that all of the necessary unique PDSS tools are acquired during the development phase as well as the development environment data bases needed.

5) The JLC (or other appropriate agencies; such as, DCA, USCG, etc.) should periodically review PDSS facilities and the systems supported by these facilities to assure that systems are supported in the most responsive and economical manner.

6) All MCCS acquisitions should be required to include the project data base and associated tools in the software development process. To promote this, a Data Item Description should be developed to define the content of the project data base and the minimum set of manipulative capabilities required.

7) PDDS's should develop management guidelines and procedures for the effective use of the project data base. Include Configuration Management, Quality Assurance, and Verification and Validation use of the project data base.

8) A clear focus of the unique performance shift between the development facility and the PDSS task should be maintained even though strong similarities now require that requisite tools and environment exist.

2.5 THE SOFTWARE CHANGE PROCESS

The purpose of this panel was to provide a uniform policy framework within which all Department of Defense support agencies function to provide efficient and timely software change support for deployed mission critical

systems. The guidelines incorporated in a " software change policy manual" are intended to cover all categories of operational equipment and attendant and/or embedded software that are furnished to the using commands and supported by DOD logistics support agencies. The Panel E report in Volume II represents the first draft of such a "software change policy manual."

2.5.1 Findings and Conclusions

It was concluded that any policy manual providing guidance for systems as diverse as submarines, tanks, and fighter aircraft must address the most generic of the required policy areas and allow reasonable flexibility in the delineation of specific structures and diverse systems requirements. Every Department of Defense agency should review the entire scope of software support required to asure systems equipment and weapon systems support taxonomies are tailored to provide reasonable economies of scale, standardization and facility utilization.

The software change process can be divided into generic segments without regard to organizational makeup or functional allocation within any particular service. Multiple services and agencies have contributed lessons learned and experiences gained to the manual. The techniques used in supporting deployed weapon systems, where significant capability is derived from the embedded software, represent the experiences gained to date and reflect only a minor subset of those anticipated to be experienced. To keep pace with the shifting support requirements (generated by the increasing knowledge base of the using and supporting commands and the on-going technological innovations), the software change process should be reviewed and updated on an annual basis.

The beginning point for all policies generated in this manual becomes a generic software change implementation model, which subdivides the change process into its fundamental discipline-based requirements areas. These are management controls, configuration management, software engineering, software quality assessment, and technical controls. In addition, the key interface areas for requirements derivation as a beginning point and user acceptance as a configuration stabilization point have been addressed. To apply this manual appropriately, it is necessary to understand that software maintenance is a term brought about by usage which distorts the understanding of the software change process itself. From an overall viewpoint, the software change model appears to be a development cycle with different acronyms and descriptions of the software development life cycle. In terms of management controls, facility requirements (including hardware and software support environments) and the operating environment context, this hardware analogy leads to a poor understanding of the extensive requirements of an efficient post deployment software support capability. Unlike the well structured requirements process which governs any major systems acquisition where significant software is involved, the software support activity is faced with the delayed accumulation of change requirements which exceed the resources available. These facts make necessary a very active management control process with significant interface to the organizations who generated the change requirements. In addition, a significant degree of real time assessment by the management control structure, including the using organization, is required to exercise decision processes geared to include or exclude specific change requests during the active portion of any on-going software change cycle.

Planning for accomplishing the software change process which must occur

during the post deployment phase should begin during the Concept Definition phase of the system life cycle as defined by DODD 5000.1. This planning will identify all internal and external interfaces, the responsible software support organization and the estimated resources (manpower, facilities, support environment and documentation) required to adequately perform PDSS.

2.5.2 Recommendations

To insure that the change process is implemented in a complete, timely, cost effective, and orderly manner, each service shall create a standard software change process based on the DOD generic software change model described in the panel report.

2.6 SOFTWARE CONFIGURATION MANAGEMENT

The role of Configuration Management (CM) in the post deployment phase of a system's life cycle is to maintain system integrity in an ever-changing environment. CM is the primary focal point for communication within the acquisition program, the support functions and the user. In the PDSS arena, CM is a continuation of the process begun in the development phase, utilizing the deliverable products as a basis for handling corrective changes, modifications an enhancements to the system's computer resources.

2.6.1 Findings and Conclusions

The following findings and conclusions emanated from the CM Panel.

1) Participation by the PDSS activity in the development phase is necessary to influence development phase configuration management practices and assure continuation of these practices into the deployment phase. The development contractor must specify PDSS parameters appropriate for support of all delivered and deployed software as part of the Software Development Plan.

2) The transition plan from developer to the Software Support Agency (SSA) should be prepared jointly and must include identification of turnover products, the schedule for delivery with contingency plans, necessary support equipment and training. The Program manager or a designated functional manager is responsible for the preparation of the Computer Resources Life Cycle Management Plan (CRLCMP) which must include a PDSS Configuration Management Plan.

3) The establishment of a DOD-wide requirement for a standard CPIN system is favored. The proposed numbering system developed by the USAF is recommended for adoption.

4) All services must store and track the same essential designated elements of configuration status accounting (CSA) information. The JLC should support the development of a common automated CSA data base system for use by all services during development and PDSS. The CSA data base should be stored in at least one location physically separate from the primary storage site.

5) Computer Resources Life Cycle Management Plans must include provisions for handling multiple, parallel baselines. An estimate of the possible extent of the situation must be included in the CRLCMP.

6) The scope of CM in PDSS must include review and identification of the impact of changes on system and subsystem interfaces as well as the integration and interoperability of interfacing systems.

2.0.2 Recommendations

The principal recommendation of the panel was that the JLC develop a PDSS configuration management policy document which requires that DOD/service directives, military standards and guidebooks relating to software management, acquisition and support that reflect the above findings and conclusions.

During panel deliberations it was determined that the software security issue during PDSS was greater in scope than could be resolved at this workshop and accordingly the following special recommendation is offered: A triservice group having expertise in the areas of hardware design, software design and support, security, configuration management and operational employment of forces be commissioned on an urgent basis to develop JLC policy recommendations and guidelines for the security aspects of current and future operational systems.

3. WORKSHOP PROCEEDINGS

The JLC Orlando I Software Workshop dealing with Post-Deployment Software Support (PDSS) is recorded in detail in Volume II of this report which comprises the workshop proceedings. In addition to the panel reports, the proceedings contain the entire workshop agenda, the workshop organization, management, administrative and technical teams.

A summary of guest speaker presentations is also included in the proceedings. The guest speakers were as follows:

1) Keynote Address by Dr. Edith W. Martin, Deputy Under Secretary of Defense for Research and Advanced Technology.

2) Luncheon Address by Dr. Robert Mathis, Technical Director of the Ada Joint Program Office in the office of the Under Secretary of Defense for Research and Advanced Technology.

3) Banquet Address by Major General Monroe T. Smith, Commander, Air Force Acquisition Logistics Division, and Chief of Staff for Acquisition Logistics, HQ Air Force Logistics Command, Wright-Patterson Air Force Base, Ohio.

4) Luncheon Address by Colonel James V. Bronson, U.S. Marine Corps, Commanding Officer, Marine Corps Tactical Systems Support Activity, MCB, Camp Pendleton, California.

5) Luncheon Address by Captain James Van Metre, U.S. Navy, Project Manager, Submarine Advanced Combat System.

The six panel reports as prepared by the panel co-chairpersons and revised after review and comment by panel members are presented in their entirety in Volume II. The findings and recommendations are presented in more detail with appropriate background and introductory information. Appendices to the panel reports include complete lists of panel members, addresses, and phone numbers, special technical papers, viewgraphs of presentations made to panels and subpanels, bibliographies and other relevant data. Each panel report is self contained in Volume II with that panels own table of contents, list of figures and list of tables.

3. WORKSHOP PROCEEDINGS

The JLC Orlando I Software Workshop dealing with Post-Deployment Software Support (PDSS) is recorded in detail in Volume II of this report which comprises the workshop proceedings. In addition to the panel reports, the proceedings contain the entire workshop agenda, the workshop organization, management, administrative and technical teams.

A summary of guest speaker presentations is also included in the proceedings. The guest speakers were as follows:

1) Keynote Address by Dr. Edith W. Martin, Deputy Under Secretary of Defense for Research and Advanced Technology.

2) Luncheon Address by Dr. Robert Mathis, Technical Director of the Ada Joint Program Office in the office of the Under Secretary of Defense for Research and Advanced Technology.

3) Banquet Address by Major General Monroe T. Smith, Commander, Air Force Acquisition Logistics Division, and Chief of Staff for Acquisition Logistics, HQ Air Force Logistics Command, Wright-Patterson Air Force Base, Ohio.

4) Luncheon Address by Colonel James V. Bronson, U.S. Marine Corps, Commanding Officer, Marine Corps Tactical Systems Support Activity, MCB, Camp Pendleton, California.

5) Luncheon Address by Captain James Van Metre, U.S. Navy, Project Manager, Submarine Advanced Combat System.

The six panel reports as prepared by the panel co-chairpersons and revised after review and comment by panel members are presented in their entirety in Volume II. The findings and recommendations are presented in more detail with appropriate background and introductory information. Appendices to the panel reports include complete lists of panel members, addresses, and phone numbers, special technical papers, viewgraphs of presentations made to panels and subpanels, bibliographies and other relevant data. Each panel report is self contained in Volume II with that panels own table of contents, list of figures and list of tables.

4. FUTURE ACTION PLAN

An "action plan" will be prepared for the JLC to implement appropriate recommendations. Workshop findings, conclusion and recommendations will be consolidated, evaluated, and prioritized forming a list for the Joint Logistics Commanders Computer Software Management Panel to consider for implementation. A schedule and assignment of responsibilities for implementing the action plan will be formulated. Items not clearly defined nor yet appropriate for action will be considered for a future study or JLC Workshop.

ATTACHMENT A:

"STRAWMAN FDSS CHARTER"

I. DESIGNATION OF ACTIVITY MANAGER

(Name of Individual) is designated as the (Name of Major Service Support Organization) Post Development Software Support Activity (PDSSA) Manager effective (Date). The PDSSA Manager reports to the Commanding General/Admiral (Service Command).

II. MISSION

The PDSSA Manager is responsible in accordance with Department of Defense (DoD) Directives (list as appropriate); Army, Navy, Air Force regulations (list as appropriate); and other pertinent regulations for:

A. Providing software life cycle support, within the scope of this charter, for all assigned systems.

B. Assessing and providing concurrence with the System Concept Paper (SCP)/Decision Coordinating Paper (DCP) and Acquisition Plan for Defense System Acquisition Review Council (DSARC) and (List corresponding service specific material acquisition decision process documentation) for adequacy of software life cycle support planning and executability.

C. Supporting the System Acquisition Manager or his/her functional representative prior to transfer of responsibility for the operational life cycle support phases.

III. AUTHORITY AND RESPONSIBILITIES

A. The Activity Manager has been delegated the full line authority of the Commanding General/Admiral Service Command for the centralized management of the (Name of Major Service Support Organization) Post Development Software Support Activity.

B. Responsibilities

1. During the concept exploration phase, the Activity Manager is responsible for advanced software support planning, including system studies to assist/advise the acquisition manager or his functional representative in specifying broad bands of software supportability and support goals/requirements. Additional responsibility includes but is not limited to:

a. Identification and planning for compliance with existing Tactical Embedded Computer Resources (TECR) policy and standardization requirements pertaining to software supportability and support.

b. Analysis of Statement of Need and other available data for potential impact on software supportability and support (threat, mission, feasibility, risk, cost, trade-offs, etc.).

c. Determination of software logistic support requirements for inclusion in the system specification (or its equivalent).

d. Preparation of a draft Software Support Plan.

e. Coordination with the Integrated Logistic Support (ILS) function.

f. Software supportability and support requirements relative to currently defined interfaces between interfacing systems and subsystems.

g. Preliminary estimate of software support cost (including acquisition of any software support resources not otherwise projected to be available, and provision of software support over the projected operational life).

2. During the demonstration and validation phase, the Activity Manager is responsible for:

a. Completing and updating the Software Support Plan.

b. Coordination with the ILS function.

c. Performing software support studies to refine and define software support requirements, including security and software logistic support requirements in particular.

d. Determining software supportability requirements to be included in software performance specification (or equivalent). Examples include reliability, modularity, programming language/Ada compiler variant, etc.

e. Updating and refining software support cost estimates.

f. Determining the requirements (types, characteristics, numbers of and availability schedule) for PDSSA equipment, to include the following types:

(1) Computers (operational; trainer; ATE; compilation; integration and test; etc.).

(2) Simulators.

(3) Selected weapon system equipment items (e.g., sensors).

g. Coordination of assignment of PDSSA functions to

the PDSSA organizational, intermediate and depot maintenance levels; contractors; and other organizations (including inter-command and inter-service organizations).

h. Estimating PDSSA support personnel requirements (types, skill levels, numbers of each) for the following:

(1) Software Engineering (test, configuration management, quality assurance, requirements definition, design, etc.).

(2) Equipment Operators (computers, simulators, etc.).

(3) Maintenance (installation of replacement computer program units or modification to in-place units; failure verification; fault isolation; checkout of installed computer programs after replacement/modification; etc.)

i. Assuring that software supportability requirements are adequately defined and put in the contract, including the contract requirement for software supportability.

3. During the Full Scale Development Phase, the Activity Manager is responsible for:

a. Technical review of the system/subsystem contractor's engineering and development effort for continued software supportability.

b. Review the developing software and related hardware configuration items (CI's) to become prepared for assuming full post deployment support responsibility. As a minimum, this should include review of all software and related hardware technical data, safety requirements and the participation in reviews and audits. In particular, the reviews should include such design elements as: functional partitioning, coding, execute/operating system, structure, data base, intermodule communications design, etc. Additionally, the software production and maintenance facility requirements, and choices of programming languages and all related support software will be included, as well as the adequacy of the contractor's quality assurance system and configuration management procedures. These reviews and any appropriate recommendations will be coordinated with the cognizant contract administration office.

c. Provide requirements to the acquisition manager or his functional representative concerning necessary equipment facilities, support software, and other material necessary to place the PDSSA software/hardware facility in full operation. Provide budgetary information for all items recommended, and obtain assistance as required from the weapon system/subsystem contractor, software developer, and other contractors to provide details and supporting information.

d. Participate in software and related hardware engineering change impact analysis as appropriate, to ensure that proposed changes do not adversely affect supportability. The PDSSA will normally continue to perform this task throughout the life cycle of the system.

e. Participate in systems contractor software T&E program through the review of test plans and procedures, as well as acting as an observer during testing. The PDSSA may provide support to technical evaluation/operational evaluation test programs as requested, and upon completion of the development phase, will normally participate directly in the acceptance testing and audit of the software and related hardware CI product baselines. These tasks are performed as an agent of the acquisition manager or his functional representative.

f. Prepare or participate in the preparation of, the weapon system computer resource life cycle management plan (CRLCMP).

g. Plan for and, as specifically directed by the acquisition manager or his functional representative, initiate action to build up facilities, equipment, and manpower (suitably trained) to the extent necessary to assume full responsibility for the system computer/processor software and related hardware support program.

h. Plan for, arrange, and conduct appropriate training for PDSSA personnel. In order to provide the capability for the PDSSA to meet all system computer/processor software and related hardware operational and support problems, and adequately support the user, extensive training is required. For major systems, experience indicates that a training period of at least two to three years is necessary. Training should begin as early in the system full-scale development phase as feasible, and on-site location training of certain PDSSA personnel at the system contractor's facility will normally be required. The detailed requirements, plans, and schedules for PDSSA buildup and training must be included in the computer resource life cycle management plan (CRLMP) and other life cycle planning documents.

i. As directed by the acquisition manager or his functional representative, participate in computer/processor software and related hardware configuration management procedures in accordance with the CRLCMP. During the later stages of the system full-scale development phase, the computer/processor system software and related hardware may undergo frequent changes to correct deficiencies which become apparent during T&E. Proper configuration management is mandatory in order to ensure validity of tests and fully define the configuration of the software and hardware that are finally delivered to the user. While this phase of configuration management normally falls under the direction of the Design Agent (DA), the PDSSA may be required by the acquisition manager or his functional representative to closely monitor the contractor's configuration management procedures during this period to ensure effectiveness and also to become thoroughly familiar with the computer/processor software and related hardware configurations. During this period, the PDSSA will develop suitable configuration management procedures for in-house service use so that

they may be activated when the PDSSA assumes full software/related hardware support responsibilities. The PDSSA configuration management procedures must comply with the service requirements and will be scheduled for implementation in accordance with the plan indicated in the CRLCMP. It is important that the PDSSA monitor configuration management, and support the software configuration review board during the full-scale development phase, so that software configuration management can be properly transitioned in accordance with the CRLCMP.

j. Conduct appropriate review of software documentation contract deliverables as they become available to determine their quality, suitability, and acceptance based upon contract requirements and their true reflection of the software being delivered. The accuracy of the software documentation is extremely important as it becomes the baseline for use by the PDSSA, T&E activities, the service, and the user as well as for future software/hardware improvements and changes. The PDSSA will develop a detailed documentation management plan which will define procedures for receipt, verification storage, duplication, distribution, inventory control, maintenance, and update.

k. Develop and prepare a detailed plan which will define procedures for assumption of responsibility for life cycle support of system computer/processor software and related hardware. This should include requirements and procedures for software inventory management, cross-indexing, storage, control, rapid retrieval, duplication, quality assurance, distribution, modification, and status accounting.

l. During the latter stages of the system full-scale development phase, a limited number of systems may be introduced to the user. The PDSSA will normally participate in user introduction at this time to prepare for assuming full responsibility in the computer/processor software and related hardware area subsequent to deployment. During this time, the PDSSA will provide liaison with users for accomplishing submittal and analysis of software trouble reports. The PDSSA will distribute updated system software and associated documentation.

m. In preparation for assuming full support responsibility, the PDSSA may participate in software/hardware problem solving in support of the DA/developer. The PDSSA may perform troubleshooting and may develop and test proposed solutions to the problem, providing such solutions to the DA/developer as an alternative problem correction.

4. During the in-service support phase of the system life cycle, the PDSSA will:

a. Assume full responsibility for life cycle support of assigned system computer/processor software and related hardware. During the in-service phase of the system, the PDSSA fulfills the requirements of a software support activity. The PDSSA will be responsible for managing the computer/processor software and ensuring that changes conform to controlled specifications, and are coordinated

with other system functional areas and managers that might be impacted. The PDSSA will ensure that computer/processor software in-service engineering support is responsive to the needs of the user. The PDSSA will perform all of the following functions.

(1) Rapid response to user software/hardware problems.
(2) Problem tracking.
(3) Problem analysis, including failure verification and fault isolation.
(4) Problem resolution and impact analysis.
(5) Development of corrections.
(6) System enhancements through software changes.
(7) Software configuration control
(8) Verification, validation, functional integration testing, and performance assurance testing
(9) Software production, distribution, and control

(10) Determine where and how installation of changes will be accomplished
(11) Software status accounting
(12) User introduction training
(13) Software documentation maintenance.

b. Be responsible for investigation of software/hardware problems and the initiation of corrective action. Prioritization of software problems and software trouble by degree of severity shall be performed. Approved software changes will be tested and verified prior to reproduction and distribution to receiving activities. These procedures will be in accordance with the information contained in the CRLCMP. Interface control documents are required to define relationships between the computer/processor system and other related systems. The PDSSA will review and recommend approval of all changes that affect these interface areas. The responsibility of the PDSSA extends to participation in problem solving at the interface level, and the testing of proposed solution that impacts the interface.

c. Assume responsibility for in-service engineering/logistics support of weapon system computer/processor software and related hardware.

d. Maintain and improve the software/hardware integration and test facility.

e. Provide continuing primary support to the acquisition manager or his functional representative and the user for assigned computer/processor software and related hardware as long as the system/subsystem remains in operation (until disposal).

IV. RESOURCE CONTROL

A. The Activity Manager will ensure that dollar and manpower

requirements to accomplish the above responsibilities are developed and submitted in accordance with established manpower/funding channels and procedures for inclusion in the Program Objective Memorandum (POM) for applicable target program years and that RDTE, procurement, operation and maintenance, and stock funds requirements are compatible at all times with the life cycle progression of assigned systems and provided in appropriate Work Breakdown Structure (WBS).

B. Monetary resources approved to accomplish the above responsibilities will be provided to the Activity Manager as direct mission funding for systems in the operational life cycle phase or by the participating organization having prime mission or task responsibility utilizing established funding channels and procedures. The Activity Manager will, in turn, provide the necessary funding, direction, or guidance, as applicable, to participating organizations for support provided in accordance wtih current regulations, policies, and procedures.

C. The Activity Manager will insure that the acquisition manager or his functional representative provides for two facilities early in the life cycle of the weapon system project: (1) A software production and maintenance facility; and (2) A software/hardware integration and test facility. These two facilities must be eventually located at, and operated by, the PDSSA.

D. PDSSA activities will ensure that the acquisition manager provides the facility with sufficient user equipment of all current versions being supported, to equip the software/hardware integration and test facility. The PDSSA facility will be considered as a field/fleet unit and will be assigned the highest Force Activity Designator justifiable under service guidelines.

V. LOCATION, SUPPORT AND STANDARDIZATION

A. Location and Support:

The (Major Service Support Organization) PDSSA is located at (Organization and Address) with necessary facilities and administrative support being provided by the organization. Liaison/field offices may be created by the Activity Manager within authorized funding as required without change of character.

B. Standardization:

The Activity Manager will:

1. Ensure that developing software systems will be designed with standardized interfaces for most efficient wartime software support and most cost effective use of established facilities and expertise.

2. Actively seek out and pursue opportunities for

promoting standardization and interoperability of assigned equipment(s) within PDSSA.

3. Incorporate interoperability requirements for all hardware and software to the maximum extent possible. (Pursue particularly electrical compatibility; mechanical interface; data and information transfer; and logistical supportability.)

4. As a minimum, review for applicability all relevant Standardization Agreements.

VI. COMMUNICATION CHANNELS

Direct communication is authorized among all participants involved in implementation of the development and support of assigned systems to ensure timely and effective direction and interchange of information among participants.