ACE-CT-100-32
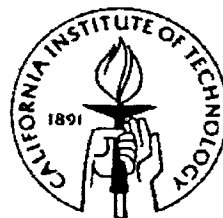
Software Implementation Plan

for the
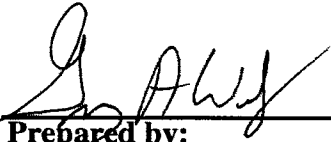
Advanced Composition Explorer Science Payload

California Institute of Technology
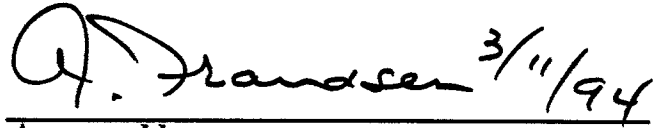
March 15, 1994

Approvals:
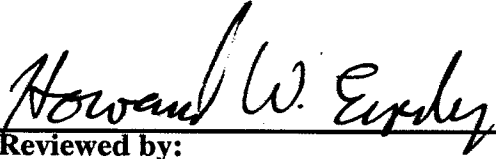
Prepared by: **_(signature)_**
G. Welz
Software Product Assurance

Approved by: **_(signature)_** 3/11/94
A. Frandsen
Payload Manager

Reviewed by: **_(signature)_**
A. Bruhn
Software Product Assurance Manager

Reviewed by: **_(signature)_**
H. Eyerly
ACE Performance Assurance Manager

ii

DOCUMENT CHANGE LOG

Revision	Date	Change Description	Preparer
Initial	March 1994	Initial Release	G. Welz

TABLE OF CONTENTS

1.0 Introduction and Background

Management of payload development for NASA's Advanced Composition Explorer (ACE) has been assigned to the California Institute of Technology (Caltech) under contract to the Goddard Space Flight Center (GSFC). Caltech is the home institution of Mission Principal Investigator Professor Edward C. Stone. Under terms of its implementation phase contract, the Caltech Payload Management Office (PMO) will establish and maintain a software management system appropriate to the development of a payload for the ACE mission.

2.0 Purpose and Scope

This document defines the PMO Software Management system and scope of the PMO software management interactions with the science payload hardware developers (SPHDs). The PMO key software management activities are described in this plan and include the following:

a. The methodology that will be employed by the PMO to accomplish the software related tasks identified in Section 10 of PMO Document ACE-CT-100-20, "Caltech Payload Assurance Implementation Plan (PAIP);"

b. PMO has obtained support from the Jet Propulsion Laboratory (JPL) Software Product Assurance organization for any necessary software performance assurance and software configuration control technical assistance;

c. Recommended methodologies that may be employed by the SPHDs to accomplish the tasks identified in Section 10 of the Caltech Payload Assurance Implementation Plan (PAIP); and

d. The software resources that PMO will make available to assist SPHDs.

3.0 Review Program

PMO will not be officially involved with software-specific SPHD reviews. Software topics will be presented at each of the PMO/GSFC co-chaired formal reviews.

3.1 Formal Reviews

The formal reviews are to be presented by the SPHDs. Formal review requirements are provided in the PAIP. The PAIP states which formal reviews are to be presented by individual SPHDs; an overview of the content of the reviews; a description of how the review results are reported and a description of the action item generation and closeout process. The four formal reviews are briefly described below. Included in the descriptions are summaries of software topics that are to be addressed at each review.

3.1.1 Inheritance Review

Inheritance Reviews will be convened for existing instruments and for previously designed instruments. The primary objectives of Inheritance Reviews is to determine if the inherited instrument (component) is appropriate for use in the ACE system and to evaluate any modifications that are necessary to adapt the inherited instrument (component) into the ACE system. Software-related Inheritance Review topics include: a description of inherited software; descriptions of hardware/software interfaces; planned software modifications; the heritage of inherited software including error histories; liens on existing software and a description of any new software builds.

3.1.2 Preliminary Design Review

Preliminary Design Reviews will be convened for newly designed instruments. The primary objective of a Preliminary Design Review is to evaluate preliminary plans before committing to final designs. Software-related Preliminary Design Review topics include: a description of software requirements and designs; descriptions of hardware/software interfaces; a description of planned software testing, and plans for software configuration management.

3.1.3 Critical Design Review

Critical Design Reviews will be convened for newly designed instruments a number of months after the Preliminary Design Reviews. The primary objective of a Critical Design Review is to evaluate final plans and designs before committing to implementation. Software-related Critical Design Review topics include: a description of software detailed designs; a description of final hardware/software interfaces; a description of software test plans, and status of software configuration management.

3.1.4 Preship Review

Preship Reviews will be convened for all instruments prior to delivery to the systems integrator. The primary objective of a Preship Review is to establish the as-delivered configuration of the instrument system; ensure that the instrument is ready for integration into the spacecraft, and identify all liens. Software related Preship Review topics include: a listing and description of all software deliverables, and identification of all software liens.

The intent of this review is to assure that the instrument is ready for integration and system testing. By the time the Preship Review has occurred, the instrument verification testing should be completed, and any problems uncovered corrected and verified.

3.1.5 Internal Reviews

PMO will recommend that internal process reviews (such as peer walk -throughs) be conducted by the SPHDs. The internal reviews should address software requirements, software design, code generation, and test plans.

4.0   Software Testing

The Caltech PMO will ensure that SPHD's tests demonstrate that the software is suitable for carrying out its required functions. Caltech plans provide for review of and assistance in generating instrument test plans, software verification and validation, software regression testing, and software configuration control.

4.1   Test Plans

The Caltech PMO will review SPHD tests plans to ensure that a methodology exists to:

a.   Verify that the nominal case executes correctly;
b.   Verify that all boundary conditions operate correctly (boundary conditions, in this context, mean those conditions where operating limits are given or implied in the requirements and design. Tests need to verify that the product operates correctly on either side of those limits. (i.e., it fails when it should, or guards against over limit situations). Another source of boundary conditions are design decisions (e.g., for fixed length arrays, what happens when the array dimension is exceeded);
c.   Verify that all failure modes operate as expected, when expected; and
d.   Verify functional or interface requirements by test cases, test cases should be explicitly derived from the requirements.

Caltech will recommend that trace matrices be incorporated into the plans, one axis for requirements and one axis for test cases. A variation on this matrix recommendation is to use a third dimension for the design modules. In this way, it becomes readily apparent what tests are to be performed on what modules and subsequently during debugging there will be a good clues indicating likely failure areas.

4.2   Regression Testing

PMO will ensure that regression testing is included as part of SPHD software test planning. During software testing, errors will undoubtedly be found and corrected; however, the corrections may have effects beyond just correcting the detected error. Therefore retesting at a higher level of previously tested modules becomes necessary. This retesting is referred to as regression testing.

4.2.1   Regression Testing Guidelines

After a correction has been inserted and verified at the unit level, re-run all of the tests or:

a.   Run the tests that uncovered the failure to ensure that the failure really was fixed;
b.   If they are available, run the diagnostic tests that were use to pin-point the error. This will help indicate any other error uncovered by this correction, and to verify this correction is correct;
c.   If time permits, run test cases to verify that nearby i.e., near operations (to the corrected operation) are also correct. This is because errors tend to occur in groups; and
d.   Plan on re-running all of test cases at least once prior to delivery, to ensure that corrections haven't broken anything else, or revealed other errors.

5.0   Software Configuration Control

5.1   Software Configuration Control

PMO will review and approve SPHD plans for controlling the configuration of software. These configuration controls will ensure that the documentation accurately describes the configured items, and that failures have been corrected and adequately treated. The plans will be in concert with software configuration control provisions contained in Section 6.0 of ACE-CT-100-031, "Configuration Management Plan for the Advanced Composition Explorer Payload." Section 6.0 of ACE-CT-100-031 states that: "The configuration of flight software will be identified by revision number and date, along with the appropriate annotation to the source code listing. The ACE Payload Change Request form will also be used by Caltech to disposition software-related requests once the flight software is placed under configuration control at the start of full-up instrument performance and acceptance testing. A duplicate of all configured code will be kept as a back-up."

5.2   Review of Software Configuration Control

PMO, with the assistance of JPL Software Product Assurance, will periodically review software products that are under configuration control. Just prior to delivery, software products and supporting documentation will reviewed at the Preship Review. Such reviews will help ensure that the configuration system is maintaining the current versions of the products, that all of the documentation agrees with the specifications, and that the documentation accurately describes the configured items.

6.0   PMO Software Development Resources

The PMO has resources available for assisting SPHDs with software development. JPL Software Product Assurance will be the primary organization supporting SPHD implementations. Reporting for these software development support tasks will be to the instrument manager or Co-Investigator. Due to resource limitations, PMO must be consulted prior to SPHDs obtaining this support from JPL. Typical support functions include:

a.   Deciding which software technical information should be kept and where it should be kept;

b.   Providing recommendations on what internal reviews, if any, should be used;

c.   The conduct of independent reviews for any technical documentation on subjects like requirements, hardware-software interfaces and test plans and procedures;

d.   Determination of if and how requirements should be traced;

e.   Assistance with the generation and verification of any requirements trace matrices created, this task is done to show that all requirements are implemented and tested;

f.   Determination of what internal configuration management procedures should be used and what is the appropriate interface between instrument development and systems integrator configuration management;

g.   Ascertaining whether or not internal error tracking/analysis should be done;

h.   The conduct of software failure modes and effects analysis and software safety analysis, (this task should only be done for software that is considered critical); and

i.   Witnessing hardware-software integration testing.

APPENDIX A

Applicable Documents

Document Number	Document Identification
ACE-CT-100-20	"Caltech Payload Assurance Implementation Plan, " Dated: December 1, 1993.
ACE-CT-100-031	"Configuration Management Plan for the Advanced Composition Explorer Payload," Dated: October 6, 1993.

APPENDIX B

Acronyms

Acronyms	Definition
ACE	Advanced Composition Explorer
Caltech	California Institute of Technology
CT	California Institute of Technology
GSFC	Goddard Space Flight Center
JPL	Jet Propulsion Laboratory
NASA	National Aeronautics and Space Administration
PAIP	Performance Assurance Implementation Plan
PMO	Payload Management Office
SPHD	Science Payload Hardware Developer