APPENDIX B

Software Development
and
Management Plan

Table of Contents

1.0 Introduction B-1
1.1 Scope B-1
1.2 Applicable Documents B-1
1.3 Acronyms and Abbreviations B-1
2.0 Overview of STEREO Mission Components B-2
3.0 Software Development Process B-3
4.0 Software Development B-3
4.1 Software Development Environment B-4
5.0 Configuration Management and Control B-4
5.1 Configuration Management B-5
5.2 Configuration Control B-5
6.0 Software Maintenance B-6

1.0 Introduction

This document describes the management approach for all software developed or procured by JHU/APL for the Solar TErrestrial RElations Observatory (STEREO) mission. Cost constraint is the primary challenge facing this mission. Therefore, the goal of this plan is to define a management approach and development guidelines that are sufficient to monitor software development and ensure the use of good engineering principles, yet not unduly burden developers with more formal reviews and documentation than are required to meet mission objectives. This plan establishes guidelines for the management, engineering, and quality assurance for the providers of STEREO software.

1.1 Scope

This management plan addresses all operational software to be developed or procured by JHU/APL for STEREO. JHU/APL has responsibility for two primary elements: the Spacecraft Bus and the Mission Operations Center. Science instruments, the Science Operations Center, and their corresponding flight and ground software are the responsibility of the Goddard Space Flight Center (GSFC).

The plan defines policy regarding providers of all software purchased, contractually acquired, developed or maintained by JHU/APL or its subcontractors for the STEREO project. It describes management mechanisms, the technical approach to the software development life cycle, required reviews and documentation, and configuration management.

1.2 Applicable Documents

(1) SRS-yy-nnn *Space Department Software Quality Assurance Guidelines 2 (Currently in Draft Form)*
(2) CLO-9805 *The Johns Hopkins University Applied Physics Laboratory Quality Assurance Plan*, August 1998
(3) APL-SDO-9989 *APL Space Department Software Quality Assurance Guidelines*; October 22, 1992.
(4) *GSFC Software Engineering Laboratory Software Development Process Guidelines*
(5) STEREO Software Quality Assurance Plan
(6) *UML Distilled*, Martin Fowler with Kendall Scott, 1997

1.3 Acronyms and Abbreviations

C&DH	Command and Data Handling
CASE	Computer Aided Software Engineering
CCB	Configuration Control Board
CDR	Critical Design Review
COTS	Commercial off-the-Shelf
DSN	Deep Space Network
ECR	Engineering Change Request
G&C	Guidance and Control
GSE	Ground Support Equipment

GSFC	Goddard Space Flight Center
I&T	Integration and Test
JHU/APL	The Johns Hopkins University Applied Physics Laboratory
MOC	Mission Operations Center
NASA	National Aeronautics and Space Administration
PDR	Preliminary Design Review
PR	Problem Report
QARL	Quality Assurance Requirement Level
SEL	Software Engineering Laboratory
SOC	Science Operations Center
SQAE	Software Quality Assurance Engineer
STEREO	Solar TErrestrial RElations Observatory
TBD	To Be Determined

2.0 Overview of STEREO Mission Components

Figure 2-1 is a block diagram showing the primary STEREO mission components.

The flight components consist of the Command and Data Handling (C&DH) system, Guidance and Control (G&C) system, and instruments for two identical spacecraft that will study solar phenomena. JHU/APL is responsible for the C&DH and G&C systems, while GSFC will select teams that will be responsible for developing the instruments.

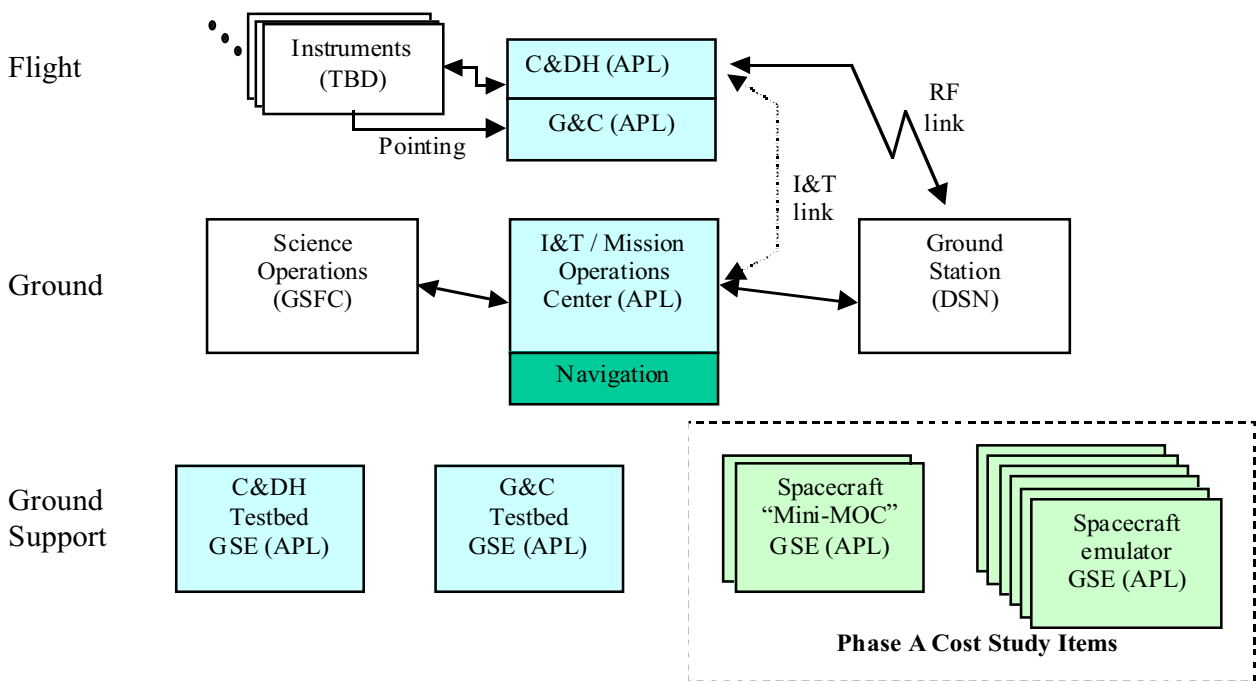

Figure 2-1. STEREO Mission Components

The major ground components consist of the Science Operations Center (SOC) to be developed and run by GSFC, the Mission Operations Center (MOC) to be developed and run by JHU/APL during spacecraft Integration and Test and flight operations, and NASA's Deep Space Network (DSN) ground stations. In addition to performing mission command, control, and monitoring functions, the MOC also includes a real time simulator for testing scenarios. JHU/APL also has a navigation team that works closely with the MOC and DSN to perform orbit determination, predictions, and updates for the mission.

In addition to the primary flight and ground system deliverables, a number of Ground Support Equipment (GSE) subsystems are required to support development and testing of deliverable hardware and software. These items, which include simulators, stimulators and bench test equipment, are generally software-based subsystems in their own right.

3.0 Software Development Process

The STEREO software development process will conform to the JHU/APL Space Department's software quality assurance guidelines described in Reference 1. The JHU/APL Space Department's Software Process Engineering Team is developing this document. The document is currently in draft form. This document will establish guidelines for software developed by the Space Department, based on its predecessor (Reference 2) and recommendations from GSFC's Software Engineering Laboratory (Reference 3).

The document outlines a process that emphasizes early coordinated analysis of the flight, ground, and GSE requirements, development of an end-to-end data architecture to meet those requirements, and an iterative design, coding, and test cycle for each subsystem. Each iteration of this cycle "builds production-quality software, tested and integrated, that satisfies a subset of the requirements". (Reference 6, p. 15) The test environment and scripts are developed in parallel with the deliverable system.

Reference 1 defines four Quality Assurance Requirement Levels (QARLs), and requires each program deliverable to be assigned a level according to its intended use (e.g., criticality). QARL 1 is the most stringent, while QARL 4 is least. All flight software and MOC software with the functions of generating and transmitting commands to the spacecraft will use QARL level 2. All other deliverable software will use QARL level 3. Overall, the STEREO program is designated to use QARL level 2.

The draft of Reference 1 that is currently available refers to enclosures that have yet to be delivered. They will provide guidance for tailoring the SEL software development process to individual programs. The final version of this software development plan will follow those guidelines as closely as possible within the cost and schedule constraints of the STEREO program.

4.0 Software Development

The software development process for STEREO will be uniform for all deliverable software components, whether they are defined as QARL 2 or 3. The primary differences will be in the formality of the documentation and the reviews required. Cost and schedule can both benefit from a standard process that emphasizes architecture and design, an adequate level of documentation, and a comprehensive review and testing program. Rather than cutting important steps from the

process in an attempt to reduce costs, STEREO will instead emphasize the use of appropriate tools to reduce the effort required in carrying out those steps.

The final version of this document will contain the identification of individual deliverable software components, their associated QARLs, and the specific documents, reviews, and tests that will be required for each.

4.1 Software Development Environment

The STEREO software development environment for flight, ground, and test software will be based on a set of support tools that assists STEREO system engineers, software architects, designers, developers and testers, and program managers in performing their tasks related to STEREO software.

The tool set will be centered on a common electronic information repository that will contain (at minimum) the following:

- a requirements database that tracks fundamental and derived requirements for all subsystems, as well as requirement dependencies between subsystems.
- current versions of software documents, including requirements specifications, architectural design documents, interface control documents, detailed design documents, user manuals, etc.
- a configuration management system and database that contains current versions and revision histories of all software, with the ability to track and recreate previous versions of full builds
- software design aids/CASE tools that help software engineers to focus on design more than coding, and to produce design documentation easily in a standard format that both developers and reviewers can understand.
- tracking and planning information such as build schedules, bug reports, etc.

The primary purposes of the software development tool set are to assist the software teams in developing quality software and to promote good communication within each subsystem team, amongst the teams, and between the software teams and program management. It is important to realize that tools require training as well as some support effort themselves, and that use of a common tool set by many development teams in different hardware environments can impose unwanted constraints on the teams. For these reasons, selection of an appropriate tool set that minimizes cost/ benefit ratios will be an important part of the software-planning phase of the project.

5.0 Configuration Management and Control

Configuration management is a process for tracking software versions, for controlling access to files in a multi-user development environment, and for documenting software development progress throughout the program. Configuration control is a process for reviewing proposed changes, assessing the costs and benefits of implementing them, approving or disapproving implementation of the changes, and documenting the decisions. The following subsections give an overview of the configuration management and configuration control mechanisms that the STEREO software team will utilize.

5.1 Configuration Management

The configuration management system/tools described in Section 4.0 will be in place before any deliverable software is developed. Software developers will receive training on using the system during the early phases of the program. Configuration management of all deliverable software files will begin at the time of creation of each file and will continue throughout the mission.

5.2 Configuration Control

The STEREO Configuration Control Board (CCB) has responsibility for managing both hardware and software configuration items. An item that has been placed under configuration control may not be modified nor granted a requirements waiver without explicit approval of the CCB. Members of the STEREO CCB include: the Mission system engineer, Mission software lead, Software I&T lead, Software quality assurance engineer and Project science representative.

STEREO software items will be placed under configuration control as follows:

Documents: Required formal documents will be placed under configuration control at the time of their sign-off. Software Requirements Specifications and software sections of Interface Control Documents will be signed off at or prior to the subsystem software PDR. Software Test Plans will be signed off before the beginning of the acceptance test phase, although nearly-complete drafts should be in place by the time of the subsystem software CDR. Documents that have been updated to reflect approved ECRs will carry a new revision letter when reissued, and will indicate the changes.

Software. Software, including COTS components, will be placed under configuration control at the start of subsystem acceptance testing, which precedes delivery of the final subsystem build to I&T. Some subsystems will include stored macros, table-driven functions and/or scriptable functions can be modified without technically making a change to application code. Any such scripts or data structures, when loaded by default upon boot or startup of the application code, shall be treated as part of the code image and will be under configuration control. The final version of this document will define the build/version-numbering scheme that provides for tracking of approved software modifications.

5.2.1 Engineering Change Request

An Engineering Change Request (ECR) may be submitted to the CCB to request a change to a configuration item. For software, the ECR includes fields to describe the requested change, the reason for the change, the scope of the change (requirements, interfaces, design, code, test), and an impact assessment including possible workarounds and testing that would be required to re-qualify the modified code. The CCB has the authority to approve or reject the ECR. The STEREO Software Quality Assurance Engineer will track the status and final disposition of all ECRs.

5.2.2 Problem Reports

A Problem Report (PR) form will be used to track apparent errors encountered during and subsequent to the I&T. For software, the PR will include fields that lists the originator, describes the problem,

the environment/circumstances in which the problem arose, and identification of the subsystem(s) involved. The PR will be routed to the cognizant engineer(s) to who will complete the form by writing a problem analysis and a recommended course of action. These courses of action include:

- none—procedural error
- fix—problem is due to software error (a description of the corrective action will be included when a "fix" is recommended)
- waiver—use operational workaround or impact is insufficient to warrant change

The PR may thus result in an ECR being written to authorize a change to a configuration item. PRs will not be considered closed out until signed off by the designated lead engineer and CCB representatives.

6.0 Software Maintenance

The CCB will continue to provide software configuration control throughout the life of the STEREO mission. Updates or modifications to any ground or flight software will be permitted only with an approved ECR. This includes updates or modifications to COTS components such as commercial databases or planning tools. Before it may be used operationally, any approved software change must undergo re-qualification with the pertinent acceptance test procedures and simulations/GSE. The Mission Operations Center will maintain a real time spacecraft simulator which will provide a means of testing modifications to flight software prior to upload to the spacecraft.