ESA PSS-05-06 Issue 1 Revision 1
March 1995

Guide to the software transfer phase

Prepared by:
ESA Board for Software
Standardisation and Control
(BSSC)

Approved by:

The Inspector General, ESA

european space agency / agence spatiale européenne
8-10, rue Mario-Nikis, 75738 PARIS CEDEX, France

DOCUMENT STATUS SHEET

DOCUMENT STATUS SHEET			
1. DOCUMENT TITLE: **ESA PSS-05-06 Guide to the Software Transfer Phase**			
2. ISSUE	3. REVISION	4. DATE	5. REASON FOR CHANGE
1	0	1994	First issue
1	1	1995	Minor updates for publication

Issue 1 Revision 1 approved, May 1995
Board for Software Standardisation and Control
M. Jones and U. Mortensen, co-chairmen

Issue 1 approved, 15th June 1995
Telematics Supervisory Board

Issue 1 approved by:
The Inspector General, ESA

Published by ESA Publications Division,
ESTEC, Noordwijk, The Netherlands.
Printed in the Netherlands.
ESA Price code: E1
ISSN 0379-4059

TABLE OF CONTENTS

This page is intentionally left blank.

PREFACE

This document is one of a series of guides to software engineering produced by the Board for Software Standardisation and Control (BSSC), of the European Space Agency. The guides contain advisory material for software developers conforming to ESA's Software Engineering Standards, ESA PSS-05-0. They have been compiled from discussions with software engineers, research of the software engineering literature, and experience gained from the application of the Software Engineering Standards in projects.

Levels one and two of the document tree at the time of writing are shown in Figure 1. This guide, identified by the shaded box, provides guidance about implementing the mandatory requirements for the software Transfer Phase described in the top level document ESA PSS-05-0.

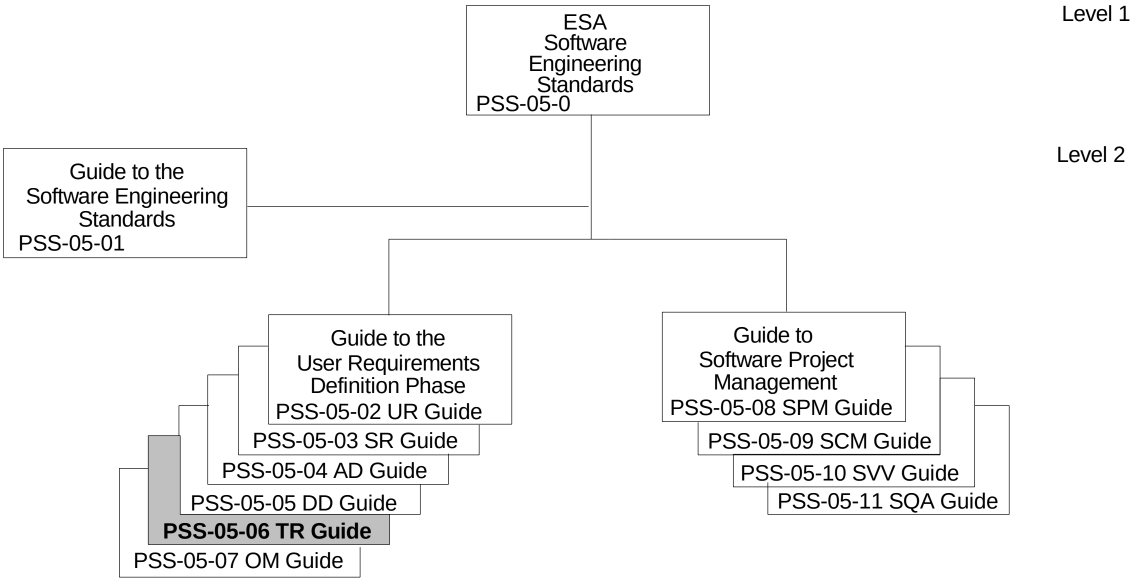

Figure 1: ESA PSS-05-0 document tree

The Guide to the Software Engineering Standards, ESA PSS-05-01, contains further information about the document tree. The interested reader should consult this guide for current information about the ESA PSS-05-0 standards and guides.

The following past and present BSSC members have contributed to the production of this guide: Carlo Mazza (chairman), Gianfranco Alvisi, Michael Jones, Bryan Melton, Daniel de Pablo and Adriaan Scheffer.

The BSSC wishes to thank Jon Fairclough for his assistance in the development of the Standards and Guides, and to all those software engineers in ESA and Industry who have made contributions.

Requests for clarifications, change proposals or any other comment concerning this guide should be addressed to:

BSSC/ESOC Secretariat
Attention of Mr M Jones
ESOC
Robert Bosch Strasse 5
D-64293 Darmstadt
Germany

BSSC/ESTEC Secretariat
Attention of Mr U Mortensen
ESTEC
Postbus 299
NL-2200 AG Noordwijk
The Netherlands

CHAPTER 1
INTRODUCTION

1.1 PURPOSE

ESA PSS-05-0 describes the software engineering standards to be applied for all deliverable software implemented for the European Space Agency (ESA) [Ref 1].

ESA PSS-05-0 defines the third phase of the software development life cycle as the ‘Detailed Design and Production Phase’ (DD phase). The outputs of this phase are the code, the Detailed Design Document (DDD) and the Software User Manual (SUM). The fourth and last phase of the software development life cycle is the ‘Transfer’ (TR) phase.

The Transfer Phase can be called the ‘handover phase’ of the life cycle because the developers release the software to the users. The software is installed on the target computer system and acceptance tests are run to validate it. Activities and products are reviewed by the Software Review Board (SRB) at the end of the phase.

This document provides guidance on how to transfer the software and write the Software Transfer Document (STD). This document should be read by everyone who participates in the Transfer Phase, such as the software project manager, software engineers, software quality assurance staff, maintenance staff and users.

1.2 OVERVIEW

Chapter 2 discusses the Transfer Phase. Chapter 3 describes how to write the Software Transfer Document.

All the mandatory practices in ESA PSS-05-0 relevant to the software Transfer Phase are repeated in this document. The identifier of the practice is added in parentheses to mark a repetition. This document contains no new mandatory practices.

This page is intentionally left blank.

CHAPTER 2
THE TRANSFER PHASE

2.1 INTRODUCTION

Software transfer is the process of handing-over the software to the users. This process requires the participation of both developers and users. It should be controlled and monitored by a test manager, who represents the users.

The Transfer Phase starts when the DD Phase Review Board has decided that the software is ready for transfer. The phase ends when the software is provisionally accepted.

The software should have been system tested in an environment representative of the target environment before the DD phase review. These system tests should include rehearsals of the acceptance tests. Such rehearsals are often referred to as Factory Acceptance Tests '(FATs).

The primary inputs to the phase are the code, Detailed Design Document (DDD) and the Software User Manual (SUM). Four plans are also required:

- Software Project Management Plan for the Transfer Phase (SPMP/TR);
- Software Configuration Management Plan for the Transfer Phase (SCMP/TR);
- Software Verification and Validation Plan, Acceptance Test section (SVVP/AT);
- Software Quality Assurance Plan for the Transfer Phase (SQAP/TR).

The Transfer Phase must be carried out according to these plans (TR03). Figure 2.1 overleaf shows the information flows into and out of the phase, and between each activity.

The first activity of Transfer Phase is to build and install the software in the target environment. The build procedures should be described in the DDD and the installation procedures should be described in the SUM. Tool support should be provided for build and installation.

The users then run the acceptance tests, as defined by the SVVP/AT. Some members of the development team should observe the tests. Test reports are completed for each execution of a test procedure. Software Problem Reports (SPRs) are completed for each problem that occurs. Tools may be employed for running the acceptance tests.

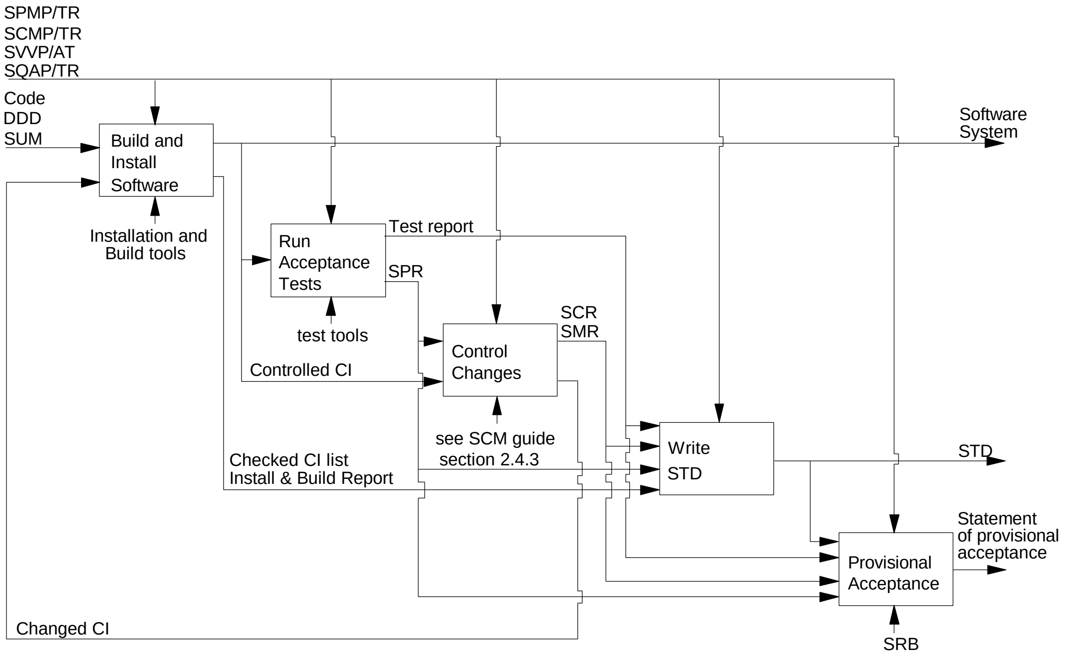

Figure 2.1: Transfer Phase activities

Problems that arise during the Transfer Phase may have to be solved before the software can be provisionally accepted. This requires the use of the change control process defined in the SCMP/TR. Software Change Requests (SCRs), Software Modification Reports (SMRs), and new versions of configuration items are produced by this process. Changed configuration items are included in a rebuild of a software. The software is then reinstalled and acceptance tests are rerun.

When problems occur, the sequence of activities of build and install software, run acceptance tests and control changes must be repeated. Software project managers should reserve enough resources to ensure that the cost of repeating these activities can be contained within the project budget.

When the acceptance tests have been completed the developer writes the Software Transfer Document (STD). This document should summarise the build and installation, acceptance test and change control activities.

When the STD is ready, the Software Review Board (SRB) meets to review the installation and acceptance test reports and any outstanding problems with the software. If the SRB is satisfied with the results, and sure that no outstanding problems are severe enough to prevent safe operation, it recommends that the initiator provisionally accepts the software. The initiator then issues a statement of provisional acceptance.

The following sections discuss in more detail the activities of software build and installation, acceptance testing, writing the STD and provisional acceptance. The change control process is discussed in Sections 2.3.3 and 2.4.3 of ESA PSS-05-09, 'Guide to Software Configuration Management '[Ref 3].

2.2 BUILD AND INSTALL SOFTWARE

Building and installing the software consists of checking the deliverables, compiling all the source files, linking the object files, and copying the resulting executable files into the target environment for acceptance tests.

The deliverables should be listed in a 'configuration item list '. A configuration item may be any type of software or hardware element, such as a document, a source file, an object file, an executable file, a disk, or a tape. The first action upon delivery is to compare the deliverables with the configuration item list. This is done by:

- checking that all the necessary media have been supplied;
- checking that the media contain the necessary files.

After the configuration check, the software is copied to where it is to be built for the target environment.

Building the software consists of compiling the source files and linking the resulting object files into executable programs. Two kinds of builds are possible:

- a complete build, in which all sources are compiled;
- a partial build, in which only selected sources are compiled (e.g those that have changed since the last build).

Partial builds often rely on the use of procedures (e.g. 'make 'files) that contain knowledge of the dependencies between files.

ESA PSS-05-0 requires that the capability of building the system from the components that are directly modifiable by the maintenance team be established (TR04). This means that the Transfer Phase must begin with a complete build of the software for the target environment. A complete build checks that all sources in the system are present, and is part of the process of checking that all the necessary configuration items are present.

The procedures for building the software should be contained in the DDD[1]. Tools should be provided for building the software, and the build procedures should describe how to operate them.

The build tools should allow both complete builds and partial builds. The ability to rapidly modify the software during the TR and OM phases may depend upon the ability to perform partial builds.

Installation consists of copying the ‘as-built’ software system to the target environment and configuring the target environment so that the software can be run as described in the Software User Manual.

The installation procedures should be described in the Software User Manual, or an annex to it.

Installation should be supported by tools that automate the process as much as possible. Instructions issued by the tools should be simple and clear. Complicated installation procedures requiring extensive manual input should be avoided.

Any changes to the existing environment should be minimal and clearly documented. Installation software should prompt the user for permission before making any changes to the system configuration (e.g. modifying the CONFIG.SYS file of a DOS-based PC).

Procedures should be provided for deinstallation, so that rollback of the system to the state before installation is possible.

2.3 RUN ACCEPTANCE TESTS

The purpose of the acceptance tests is to show that the software meets the user requirements specified in the URD. This is called ‘validation’.

[1] ESA PSS-05-0 Issue 2 omitted this section from part one of the DDD. This section will be added to the DDD template in the next issue of the ESA Software Engineering Standards.

The acceptance tests are performed on the software system installed in the previous stage of the Transfer Phase.

This section addresses the following questions that have to be considered when validating the software:

- where are the acceptance tests specified?
- who should participate?
- what tools are suitable?
- what criteria should be used for passing the tests?
- what criteria should be used for suspending the tests?

2.3.1 The Acceptance Test Specification

The acceptance tests are specified in the acceptance test section of the Software Verification and Validation Plan (SVVP/AT). The SVVP/AT contains five subsections, the first four of which are prepared before the Transfer Phase:

- test plan;
- test designs;
- test cases;
- test procedures;
- test reports.

The test plan is prepared as soon as the User Requirements Document (URD) is completed at the end of the UR phase (SVV11). The test plan outlines the scope, approach, resources and schedule of the acceptance tests.

In the DD phase, tests are designed for the user requirements (SVV19), although some acceptance test design may be done earlier. Test cases and procedures are defined in the DD phase when the detailed design of the software has sufficiently stabilised (SVV20, SVV21).

The acceptance tests necessary for provisional acceptance (see Section 2.5) must be indicated in the SVVP (TR05). Some of the tests (e.g. of reliability) often require an extended period of operations, and can only be completed by final acceptance. Such tests should be clearly identified.

Personnel that carry out the tests should be familiar with the SUM and the operational environment. Instructions in the SUM need not be repeated in the SVVP.

Acceptance test reports are prepared in the Transfer Phase as each acceptance test is performed. They should be added to the SVVP/AT (SVV22). Possible formats include:

- acceptance test report forms recording the date and outcome of the test cases executed by the procedure;
- execution log file.

Guidance on the preparation of the SVVP/AT is contained in Guide to Software Verification and Validation', ESA PSS-05-10 [Ref 4]. These guidelines will be of interest to developers, who have to prepare the SVVP/AT. The following sections deal with acceptance test issues relevant to both users and developers.

2.3.2 Participation in Acceptance Tests

Users may be end users (i.e. a person who utilises the products or services of the system) or operators (i.e. a person who controls and monitors the hardware and software of a system). Both end users and operators must participate in the acceptance tests (TR01). A test manager should be appointed to control and monitor the acceptance tests and represent the users.

Members of the development team and an SQA representative should participate in the acceptance tests.

Acceptance testing is not a substitute for a training programme, which should be scheduled separately.

2.3.3 Acceptance Test Tools

Acceptance test procedures should be automated as far as possible. Tools that may be useful for acceptance testing are:

- test harnesses for running test procedures;
- performance analysers for measuring resource usage;
- comparators for comparing actual results with expected results;
- test management tools for organising the test data and scripts.

Simulators should be used when other parts of the system are not available or would be put at risk.

2.3.4 Acceptance Test Pass and Fail Criteria

The acceptance test plan (SVVP/AT/Plan) defines the test pass and fail criteria. These should be as objective as possible. The criterion that is normally used is the software will be judged to have passed an acceptance test if no critical software problems that relate to the test occur during the Transfer Phase ’. Projects should define criteria for criticality, for example:

- critical: crash, incorrect result, unsafe behaviour;
- non-critical: typing error, failure to meet a non-essential requirement.

2.3.5 Acceptance Test Suspension and Resumption Criteria

The acceptance test plan defines the test suspension and resumption criteria. The suspension criteria should define the events that cause the abandonment of an acceptance test case, an acceptance test procedure and the acceptance test run. Criteria may, for example, be based upon test duration, percentage of tests failed, or failure of specific tests.

Flexibility in the application of the suspension criteria is usually necessary. For example a single critical software problem might cause a test procedure or even a whole test run to be abandoned. In other cases it may be possible to continue after experiencing problems, although it may not be an efficient use of resources to do so. In some situations, it may be possible to complete the acceptance tests even if there are several problems, because the problems lie within a subsystem that can be tested independently of the rest of the system.

When the problems that have caused the abandonment of an acceptance test procedure have been corrected, the entire test procedure should be repeated. All the acceptance test procedures should normally be repeated if acceptance tests are resumed after the abandonment of a test run. The decision about which acceptance tests are rerun lies with the users, in consultation with the developers.

2.4 WRITE SOFTWARE TRANSFER DOCUMENT

The purpose of the Software Transfer Document (STD) is to present the results of the Transfer Phase in a concise form, so that the Software

Review Board can readily decide upon the acceptability of the software. Further information about the STD is contained in Chapter 3.

2.5 PROVISIONAL ACCEPTANCE

When the acceptance tests have been completed, the software is reviewed by the SRB. Inputs to the review are the:

- Software Transfer Document (STD);
- acceptance test results;
- Software Problem Reports made during the Transfer Phase;
- Software Change Requests made during the Transfer Phase;
- Software Modification Reports made during the Transfer Phase.

The Software Review Board (SRB) reviews these inputs and recommends, to the initiator, whether the software can be provisionally accepted or not (TR02).

The responsibilities and constitution of the SRB are defined in Section 2.2.1.3 of ESA PSS-05-09 ‘Guide to Software Configuration Management ’ [Ref 3]. The software review should be a formal technical review. The procedures for a technical review are described in Section 2.3.1 of ESA PSS-05-10, ‘Guide to Software Verification and Validation ’[Ref 4].

The SRB should confirm that the deliverables are complete and up to date by reviewing the checked configuration item list, which is part of the Software Transfer Document. The SRB should then evaluate the degree of compliance of the software with the user requirements by considering the number and nature of:

- failed test cases;
- tests cases not attempted or not completed;
- critical software problems reported;
- critical software problems not solved.

Solution of a problem means that a change request has been approved, modification has been done, and all tests repeated and passed.

The number of test cases, critical software problems and non-critical software problems should be evaluated for the whole system and for each subsystem. The SRB might, for example, decide that some subsystems are acceptable and others are not.

When the number of problems encountered in acceptance tests is low, the SRB should check the thoroughness of the acceptance tests. Experience has indicated that the existence of faults in delivered software is the rule rather than the exception (typically 1 to 4 defects per 1000 lines of code [Ref 5]).

Where appropriate, the SRB should examine the trends in the number of critical software problems reported and solved. There should be a downward trend in the number of critical problems reported. An upward trend is a cause for concern.

If the SRB decides that the degree of compliance of the software with the user requirements is acceptable, it should recommend to the initiator that the software be provisionally accepted.

The statement of provisional acceptance is produced by the initiator, on behalf of the users, and sent to the developer (TR06). The provisionally accepted software system consists of the outputs of all previous phases and the modifications found necessary in the Transfer Phase (TR07). Before it can be finally accepted, the software must enter the Operations and Maintenance phase and undergo a period of operations to prove its reliability.

This page is intentionally left blank.

CHAPTER 3
THE SOFTWARE TRANSFER DOCUMENT

3.1 INTRODUCTION

ESA PSS-05-0 states that the Software Transfer Document (STD) shall be an output of the Transfer Phase (TR08). The purpose of the STD is to summarise:

- the software that has been delivered;
- the build of the software;
- the installation of the software;
- the acceptance test results;
- the problems that occurred during the Transfer Phase;
- the changes requested during the Transfer Phase;
- the modifications done during the Transfer Phase.

3.2 STYLE

The STD should be concise, clear and consistent.

3.3 EVOLUTION

The developer should prepare the STD after the acceptance tests and before the software review.

If the software is provisionally accepted, the completed STD must be handed over by the developer to the maintenance organisation at the end of the Transfer Phase (TR09).

After the Transfer Phase the STD should not be modified. New releases of the software in the Operations and Maintenance Phase are documented in Software Release Notes (SRNs), rather than in new issues of the STD.

3.4 RESPONSIBILITY

The developer is responsible for the production of the STD. The software librarian is a suitable person to write it.

3.5 MEDIUM

The STD is normally a paper document.

3.6 CONTENT

ESA PSS-05-0 recommends the following table of contents for the Software Transfer Document:

1 Introduction
  1.1 Purpose
  1.2 Scope
  1.3 Definitions, acronyms and abbreviations
  1.4 References

2 Build Procedures[2]

3 Installation Procedures

4 Configuration Item List

5 Acceptance Test Report Summary

6 Software Problem Reports

7 Software Change Requests

8 Software Modification Reports

[2] ESA PSS-05-0 Issue 2 states that Build Procedures must be section 3 and Installation Procedures must be Section 2. This order will be reversed in the next issue of the ESA Software Engineering Standards.

3.6.1 STD/1 INTRODUCTION

3.6.1.1 STD/1.1 Purpose

This section should define the purpose of the STD, for example: ‘This Software Transfer Document (STD) reports in summary form the Transfer Phase activities for the [insert the name of the project here].’

3.6.1.2 STD/1.2 Scope

This section should summarise the software that is being transferred.

Projects that adopt incremental delivery or evolutionary life cycle approaches will have multiple Transfer Phases. Each Transfer Phase must produce an STD. Each STD should define the scope of the software being transferred in that particular Transfer Phase.

3.6.1.3 STD/1.3 Definitions, acronyms and abbreviations

This section should define all the terms, acronyms, and abbreviations used in the STD, or refer to other documents where the definitions can be found.

3.6.1.4 STD/1.4 References

This section should provide a complete list of all the applicable or reference documents, identified by title, author and date. Each document should be marked as applicable or reference. If appropriate, report number, journal name and publishing organisation should be included.

3.6.2 STD/2 BUILD PROCEDURES

This section should report what happened when the software system was built. For example this section may report:

- when the software was built;
- where the software was built;
- the hardware and software environment in which the software was built;
- which version of the software was built;
- any problems encountered during the build;
- the CPU time and elapsed time required for building the system;
- the elapsed time required for building the system.

This section should reference the build procedures, which should have been defined in the Detailed Design Document[3].

3.6.3 STD/3 INSTALLATION PROCEDURES

This section should report what happened when the software was installed. For example this section may report:

- when the software was installed;
- where the software was installed;
- the hardware and software environment in which the software was installed;
- which version of the software was installed;
- any problems encountered during the installation;
- the elapsed time required for installing the system;
- the disk space used;
- the system parameters modified.

The organisation of the software files in the target environment should be described (e.g. root directory, subdirectories etc).

[3] ESA PSS-05-0 Issue 2 omitted this section from part one of the DDD. This section will be added to the DDD template in the next issue of the ESA Software Engineering Standards.

This section should reference the installation procedures, which should have been defined in the Software User Manual, or an annex to it.

3.6.4 STD/4 CONFIGURATION ITEM LIST

This section should contain the list of the configuration items transferred. There should be evidence that the transferral of each configuration item has been verified (e.g. a completed checklist).

3.6.5 STD/5 ACCEPTANCE TEST REPORT SUMMARY

This section must contain a summary of the acceptance test reports (TR10). An overall statement of pass or fail for each subsystem should be provided. Specific test failures should be identified and discussed.

This section should identify all the user requirements that the software does not comply with. The list should include the value of the need attribute of the requirement (e.g. essential, desirable).

This section should present data for the following metrics:

- number of acceptance test cases passed;
- number of acceptance test cases failed;
- number acceptance test cases not attempted or completed.

Data should be presented for the whole system and individually for each subsystem.

3.6.6 STD/6 SOFTWARE PROBLEM REPORTS

This section should list the Software Problem Reports (SPRs) raised during the Transfer Phase and their status at STD issue. Information about software problems should be present in the project configuration status accounts.

This section should present data for the following metrics:

- number of software problems reported;
- number of software problems closed;
- number of software problems open (i.e. status is action or update).

Data should be presented for all the SPRs, critical SPRs and non-critical SPRs.

3.6.7 STD/7 SOFTWARE CHANGE REQUESTS

This section should list the Software Change Requests (SCRs) raised during the Transfer Phase and their status at STD issue. Information about software changes should be present in the project configuration status accounts.

This section should present data for the following metrics:

- number of software change requests closed;
- number of software change requests open (i.e. status is action or update).

Data should be presented for all the SCRs, critical SCRs and non-critical SCRs.

3.6.8 STD/8 SOFTWARE MODIFICATION REPORTS

This section must list the Software Modification Reports (SMRs) completed during the Transfer Phase (TR10). Information about software modifications should be present in the project configuration status accounts.

This section should present data for the following metrics:

- number of software modifications closed;
- number of software modifications in progress.

Data should be presented for all the SMRs, critical SMRs and non-critical SMRs. An SMR is critical if it relates to a critical SCR.

APPENDIX A
GLOSSARY

A.1 LIST OF TERMS

Terms used in this document are consistent with ESA PSS-05-0 [Ref 1] and ANSI/IEEE Std 610.12 [Ref 2]. Additional terms not defined in these standards are listed below.

complete build

The process of compiling all the source files written by the developers and linking the object files produced with each other, and external object files.

development environment

The environment of computer hardware, operating software and test software in which the software system is developed.

end user

A person who utilises the products or services of a system.

installation

The process of copying a software system into the target environment and configuring the target environment to make the software system usable.

operational environment

The target environment, external software and users.

operator

A person who controls and monitors the hardware and software of a system.

partial build

The process of compiling selected source files (e.g those that have changed since the last build) and linking the object files produced with each other, and pre-existing object files.

target environment

The environment of computer hardware and operating software in which the software system is used.

user

A person who utilises the products or services of a system, or a person who controls and monitors the hardware and software of a system (i.e. an end user, an operator, or both).

A.2 LIST OF ACRONYMS

ANSI	American National Standards Institute
AT	Acceptance Test
BSSC	Board for Software Standardisation and Control
CASE	Computer Aided Software Engineering
DD	Detailed Design and production
FAT	Factory Acceptance Test
SCMP	Software Configuration Management Plan
SCR	Software Change Request
SMR	Software Modification Report
SPMP	Software Project Management Plan
SPR	Software Problem Report
SQAP	Software Quality Assurance Plan
SR	Software Requirements definition
SRD	Software Requirements Document
SRN	Software Release Note
STD	Software Transfer Document
SVVP	Software Verification and Validation Plan
TR	Transfer
UR	User Requirements definition
URD	User Requirements Document

This page is intentionally left blank.

APPENDIX B
REFERENCES

1. ESA Software Engineering Standards, ESA PSS-05-0 Issue 2 February 1991.

2. IEEE Standard Glossary of Software Engineering Terminology, ANSI/IEEE Std 610.12-1990

3. Guide to Software Configuration Management, ESA PSS-05-09, Issue 1, November 1992

4. Guide to Software Verification and Validation, ESA PSS-05-10, Issue 1, December 1993

5. Managing Computer Projects, R. Gibson, Prentice-Hall, 1992.

This page is intentionally left blank

APPENDIX C
MANDATORY PRACTICES

This appendix is repeated from ESA PSS-05-0 appendix D.6

TR01 Representatives of users and operations personnel shall participate in acceptance tests.

TR02 The Software Review Board (SRB) shall review the software's performance in the acceptance tests and recommend, to the initiator, whether the software can be provisionally accepted or not.

TR03 Transfer Phase activities shall be carried out according to the plans defined in the DD phase.

TR04 The capability of building the system from the components that are directly modifiable by the maintenance team shall be established.

TR05 Acceptance tests necessary for provisional acceptance shall be indicated in the SVVP.

TR06 The statement of provisional acceptance shall be produced by the initiator, on behalf of the users, and sent to the developer.

TR07 The provisionally accepted software system shall consist of the outputs of all previous phases and modifications found necessary in the Transfer Phase.

TR08 An output of the Transfer Phase shall be the STD.

TR09 The STD shall be handed over from the developer to the maintenance organisation at provisional acceptance.

TR10 The STD shall contain the summary of the acceptance test reports, and all documentation about software changes performed during the Transfer Phase.

This page is intentionally left blank

APPENDIX D
INDEX