LARGE SYNOPTIC SURVEY TELESCOPE

PROD Software Test Plan

prepared by: theAuthor
approved by: theApprover

reference: GAIA-C?-SP-???-???-???-0D
issue: 0D
revision: 1
date: 2017-01-14
status: draft

Abstract

THERE IS ONLY ONE OF THESE YOU PROBABLY WANT AN STS This is a template for the Gaia DPAC Software Test Plan for PROD. It applies to Cycle CYC-NUM

Document History

Issue	Revision	Date	Author	Comment
D	1	yyyy-mm-dd	WOM	First draft

Contents

1 Introduction

THERE IS ONLY ONE OF THESE YOU PROBABLY WANT AN STS

1.1 Objectives

The Software Test Plan describes the system being tested, summarising the system context and decomposition. It sets out the test and verification approach for the system and describes constraints and limitations in the testing to be performed. The STP describes the unit and integration tests for the component modules of the system and describes the validation tests to be performed on the fully integrated system.

1.2 Scope

The Software Test Plan is to be executed by the CU prior to delivery to the DPC where the system will be operated. The DPC will execute integration and acceptance test involving this system within the context of the DPC processing systems. This document will be updated during the different Gaia cycle phases, according to the requirements updates.

1.3 Assumptions

This paragraph is optional. It describes the preliminary assumptions on which the overall testing strategy are based.

1.4 Applicable Documents

When applicable documents change a change may be required in this document.

TL-001	DPAC Product Assurance Plan
?	Software Development Plan for DM
?	Software Requirements Specification for PROD,

1.5 Reference Documents

[1] **[TL-001]**, Levoir, T., Damery, J., Hoar, J., et al., 2012, *DPAC Product Assurance Plan*, GAIA-C1-PL-CNES-TL-001,
URL http://www.rssd.esa.int/cs/livelink/open/2439085

1.6 Definitions, acronyms, and abbreviations

The following table has been generated from the on-line Gaia acronym list:

Acronym	Description
AGIS	Astrometric Global Iterative Solution
AO	Announcement of Opportunity
CCB	Configuration Control Board
CM	Configuration Management
CU	Coordination Unit (in DPAC)
DOC	Department of Commerce (USA)
DPAC	Data Processing and Analysis Consortium
DPC	Data Processing Centre
DU	Detection Unit
ECSS	European Cooperation for Space Standardisation
ESA	European Space Agency
ESAC	European Space Astronomy Centre (VilSpa)
GWP	Gaia Work Package
PA	Product Assurance
QA	Quality Assurance
SCMP	Software Configuration Management Plan
SRS	Software Requirements Specification
SW	Software (also denoted S/W)
TOC	Table of Contents
WBS	Work Breakdown Structure
WP	Work Package

2 Test Items

The test items covered in this test plan are PROD and its consituent components:

- *Name and brief description of Component A*
- *Name and brief description of Component B*
- *Name and brief description of Component C*

3 Roles and Reporting

Tester report issues through Mantis, but what other mechanisms will be used?

3.1 Pass/Fail Criteria

The Software Review Board will meet once a full run of all Test Cases has been performed, and subsequently after a complete run of all outstanding Test Cases.

A Test Case will be considered “Passed” when:

- All of the test steps of the Test Case are completed and
- All open SPRs from this Test Case agreed in Software Review Board are considered noncritical.

A Test Case will be considered “Partially Passed” when:

- Only a subset of all of the test steps in the Test Case are completed but the overall purpose of the test has been met and
- Any critical SPRs from this Test Case agreed in Software Review Board are still not closed.

A Test Case will be considered “Failed” when:

- Only a subset of all of the test steps in the Test Case are completed and the overall purpose of the test has not been met and
- Any critical SPRs from this Test Case agreed in Software Review Board are still not closed.

4 Constraints and Limitations

Describes the limitations and the constraints which apply to CU level tests of the system. lack of computing resources may mean that datasets are smaller or that full accuracy cannot be achieved. Explain what must be validated in the DPC tests

5 Master Schedule

The schedule for testing the system until launch. If some modules are scheduled for development after other, explain dependencies and impact on integration and validation tests.

6 Validation Tools

6.1 Introduction

To evaluate the correctness of the generated data and the systems performances a set of tools may be developed or used. These tools will provide the means to facilitate the validation tasks. Following subsections describe the various tools that can used in the PRODvalidation (e.g. data comparison tools, analysis tools, etc).

6.2 Data Comparison Tools

This type of test tools are used to manage products in terms of:

- Comparison of a product generated during a test execution w.r.t. the relevant reference product
- Non regression verification comparing output products generated by different versions of the same system
- Measurement of quality degradation due to perturbed inputs

It allows:

- Product analysis
- Decoding of generated product allowing to read the most significant data of the product itself
- Visualisation of the values of a single selected field
- Apply an accuracy to the comparison

- Comparing specific parts of the products
- Filtering using flags values

6.3 Data Transformation Tools

These tools allow the data to be transformed to other formatted data.

6.4 Analysis Tools

Descriptions of the performance monitoring tools, profilers, test coverage programs... used in the Performance evaluation tests.
...

7 Unit and Integration Tests

7.1 Approach

Unit and Integration Tests will be automatically executed through the JUnit test framework. The descriptions of the test below are extracted from the test cases code and documentation.The results of Unit and Integration Test to be included in the Sofwtare Test Report will be generated automatically from the output of the execution of the tests by JUnit. A script will be provided to perform thes processing steps.

Module identification? (module tag in class header? mapping file?)

7.2 Test Coverage

Test coverage goal for unit and integration testing. Each class and public method shall have a JUnit test harness that may be labelled according to their purpose (e.g. I/O, individual class/method tests, software integration, data model integration etc.). Nominal and contingency tests should be clearly identified.

Interface coverage...

The tool [insert name of unit test coverage tool here] will be used to provide metrics on the code coverage by Unit Tests for PROD and this metric will be provide in the Test Report.

7.3 Unit and Integration Test Specification

This is a example test plan record; this should be generated automatically.

Class	Unit Test Name	Purpose
Unit Test Class	Unit Test Method	Purpose of Unit Test from method header

8 Validation Tests

8.1 General strategy

Description of the general verification and validation strategy, decomposition into verification testing categories (e.g. science tests, SP external interface tests, algorithms interrelation and sequence). Assessed validation tests results shall be available over the software development duration: they are stored into SVN repository along with related input data, property-file, etc.

A subset of tests are run at DPC during software release qualification process, the results of DPC runs are compared with corresponding test outputs. During DPC integration tests, these assessed outputs will also allow to verify software non-regression.

8.2 Test Designs

8.2.1 Test Design DM-PROD-SYS-X

8.2.1.1 Objective

Explain the objective of this test design

8.2.1.2 Features to be tested

- Component A
- Component B

8.2.1.3 Features not to be tested

- Component C
- Component D

8.2.1.4 Approach Description of the approach to writing this test design

8.2.1.5 Test Cases List of test cases to be specified

Test Case	Description
DM-PROD-SYS-X-1	Description of Validation Test

8.3 Test Case Specification

8.3.1 Test Case DM-PROD-SYS-X-1

8.3.1.1 Test Items List the components to be tested in this test case

8.3.1.2 Purpose Explain the purpose of this test case

8.3.1.3 Input Specification Describe the inputs to this test (data, written procedures, etc.)

8.3.1.4 Output Specification Describe the outputs of this test

8.3.1.5 Environment Describe the environment (computing resources etc) required for this test.

8.3.1.6 Inter-case dependencies If this test in dependent on another test having been completed successfully (for input data for example), state that here.

8.3.1.7 Test Procedure Describe the procedure to be performed

8.3.1.8 Test Verification Describe how to verify if the test has been successful.

8.4 Traceability to Requirements

The traceability between the Requirements describing this system (at SRS or higher level) and the Validation Test Cases should be given here. A script will be provided to create this.